Частное учреждение образования   
«Колледж бизнеса и права»

ПРОГРАММА ДЛЯ АВТОМАТИЗАЦИИ УЧЁТА ИНФОРМАЦИИ О ПРИВИВКАХ И НАГРАДАХ ЖИВОТНЫХ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту по дисциплине

«Базы данных и системы управления базами данных»

КП Т.593005.401

Учащийся (Д.А. Зеленко )

Руководитель проекта (С.В. Банцевич)

2018

Содержание

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

КП Т.593005.401 ПЗ

Разраб.

Зеленко Д.А.

Провер.

Банцевич С.В.

Реценз.

Н. контр.

Утверд.

*Программа для автоматизации учёта информации о прививках и наградах животных*

Лит.

Листов

КБП

0

у

[Введение 4](#_Toc531286173)

[1 Объектно-ориентированный анализ и проектирование системы 5](#_Toc531286174)

[1.1 Сущность задачи 5](#_Toc531286175)

[1.2 Проектирование модели 7](#_Toc531286176)

[2 Вычислительная система 13](#_Toc531286177)

[2.1 Требования к аппаратным и операционным ресурсам 13](#_Toc531286178)

[2.2 Инструменты разработки 13](#_Toc531286179)

[3 Проектирование задачи 15](#_Toc531286180)

[3.1 Требования к приложению 15](#_Toc531286181)

[3.2 Концептуальный прототип 15](#_Toc531286182)

[3.3 Организация данных 17](#_Toc531286183)

[3.4 Функции и элементы управления 22](#_Toc531286184)

[3.5 Проектирование справочной системы 23](#_Toc531286185)

[4 Описание программного средства 24](#_Toc531286186)

[4.1 Общие сведения 24](#_Toc531286187)

[4.2 Функциональное назначение 24](#_Toc531286188)

[4.3 Входные и выходные данные 24](#_Toc531286189)

[5 Методика испытаний 29](#_Toc531286190)

[5.1 Технические требования 29](#_Toc531286191)

[5.2 Функциональное тестирование 29](#_Toc531286192)

[6 Применение 36](#_Toc531286193)

[6.1 Назначение программы 36](#_Toc531286194)

[6.2 Условия применения 36](#_Toc531286195)

[6.3 Справочная система 36](#_Toc531286196)

[Заключение 38](#_Toc531286197)

[Список информационных источников 39](#_Toc531286198)

[Приложение А Текст программных модулей 40](#_Toc531286199)

# Введение

Выставка животных – конкурс зрелищный и зоотехнический, мероприятие, на котором представлены различные породы животных. Назначение выставочного показа – по возможности собрать представителей породистого мира.

Благодаря такому конкурсу можно получить представление о поголовье породистого класса животных на конкретной территории. Выставка показывает направленность потомков, не только выявить самых лучших породистых представителей помогает конкурсный показ.

Целью курсового проекта для автоматизации учёта информации о прививках и наградах животных является разработка программного средства «ZelZooExpo.exe», которая будет осуществлять ведение базы данных, содержащей информацию о животных, владельцах, наградах, прививках, паспортных данных животного, осуществлять контроль календаря прививок, генерировать различные отчёты.

Пояснительная записка состоит из шести разделов, содержащих необходимую информацию по организации эксплуатации программного средства.

В первом разделе «Объектно-ориентированный анализ и проектирование системы» раскрывается организационная сущность задачи, описывается предметная область и круг задач, которые должны быть автоматизированы. Описывается задача, перечисляются основные функции программы. Строится информационная модель, отражающая сущности задачи, их свойства и взаимосвязи.

Во втором разделе «Вычислительная система» перечисляются требования к аппаратному и программному обеспечению компьютера, приводится характеристика операционной системы, обоснование выбранной среды для разработки программы.

В третьем разделе «Проектирование задачи» проводится объектно-ориентированный анализ задачи, приводится требования к приложению, концептуальный прототип, организация данных, функции и элементы управления, проектирование справочной системы приложения.

В четвертом разделе «Описание программного средства» представлены общие сведения о программном средстве и его функциональном назначении, входные и выходные данные.

В пятом разделе «Методика испытаний» описываются требования к аппаратному и программному обеспечению компьютера для проведения испытаний, требования к характеристикам программы применительно к условиям эксплуатации, требования к информационной и программной совместимости. Представляются результаты функционального тестирования.

Шестой раздел «Применение» предназначен для описания сведений о назначении программного средства и области его применения. В этом разделе приводится структура справочной системы, а также методика ее использования.

В заключении анализируется созданное программное средство, определяется степень соответствия поставленной задачи и выполненной работы.

Приложение А содержит текст программных модулей.

В графической части представлены диаграммы вариантов использования, классов, деятельности, последовательности и компонентов.

# Объектно-ориентированный анализ и проектирование системы

## Сущность задачи

Предметной областью решаемой задачи является организация «ZelZooExpo» по проведению выставок животных.

Выставка животных – публичное мероприятие по оценке соответствия животных стандартам определённой породы с целью их дальнейшего разведения.

Организации для проведения выставки нужно собрать различные данные, такие как: паспорт животного, ветеринарное свидетельство животного, предыдущие награды животного, данные владельца.

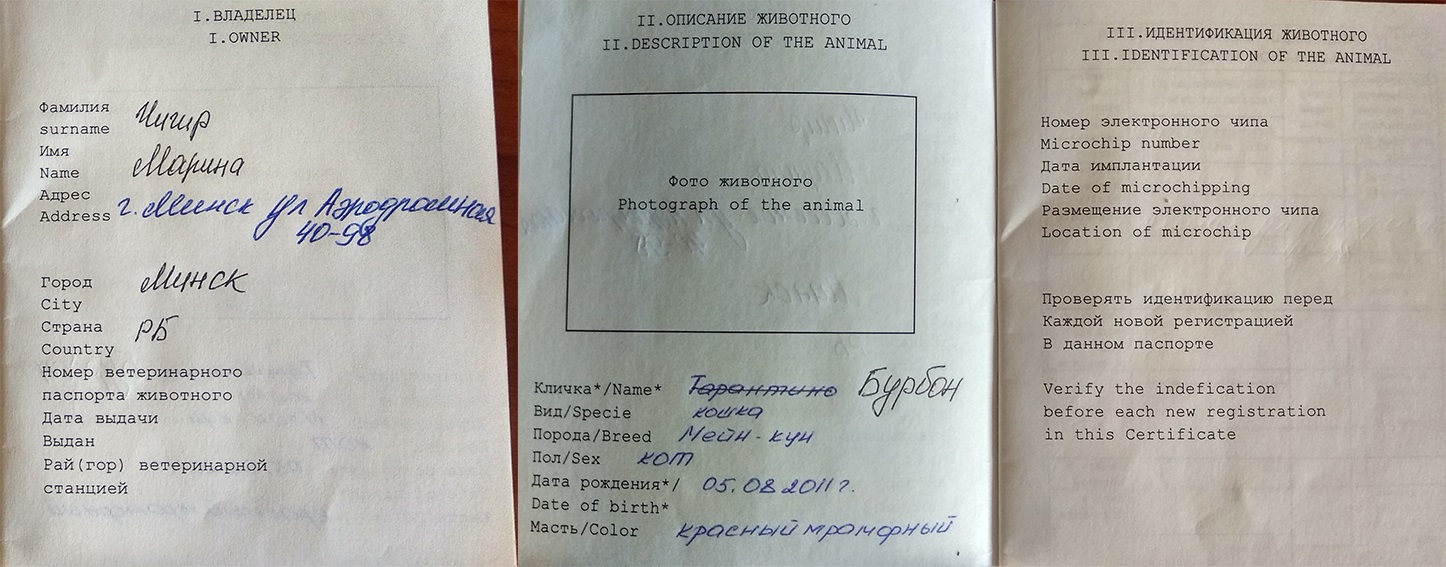
Для того чтобы животное могло принять участие в выставке, необходимо чтобы у него были сделаны все необходимые прививки.

**Паспорт животного представлен на рисунке 1.1**



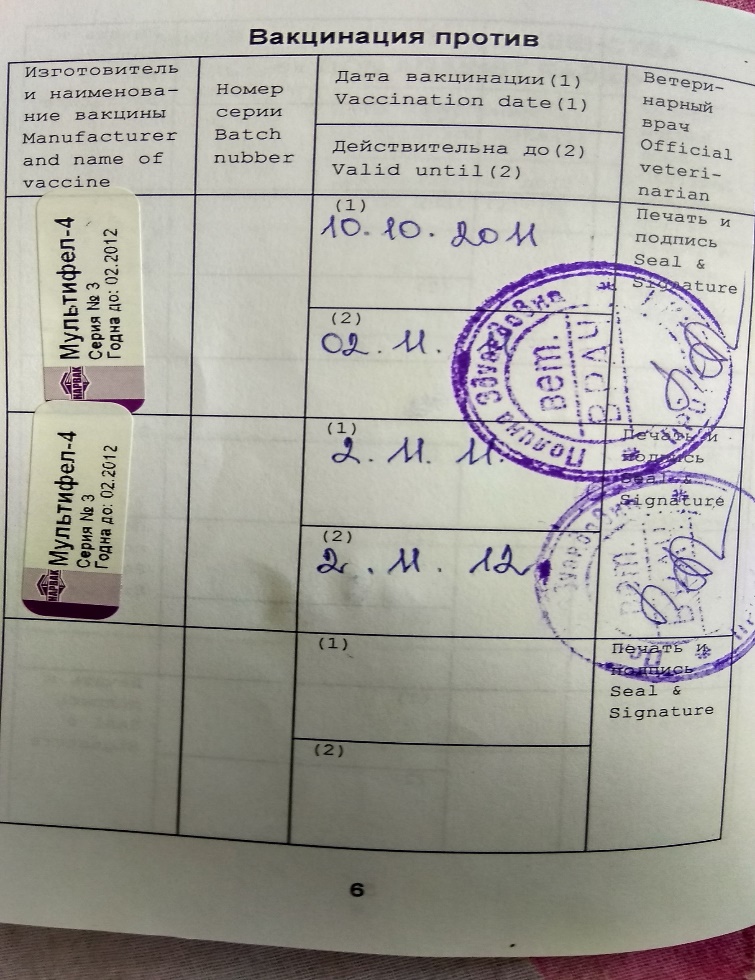
**Рисунок 1.1**

**Ветеринарное свидетельство представлено на рисунке 1.2**



**Рисунок 1.2**

**Календарь прививок представлен на рисунке 1.3**



**Рисунок 1.3**

**Награда животного представлена на рисунке 1.4.**



**Рисунок 1.4**

Исходя из анализа предметной области можно выделить следующие задачи, подлежащие автоматизации:

– ведение базы данных (БД) животных, о их прививках, наградах, владельцах, странах, породах, о месте размещения электронного чипа, ветеринарном свидетельстве;

– осуществление контроля и актуализации сведений о характеристиках животных, ветеринарных свидетельств, наградах;

– хранение графического материала;

– контроль соблюдения сроков календаря прививок;

– просмотр и поиск необходимой пользователю информации; сортировка данных по различным критериям;

– создание отчётов по интересующей пользователя информации: ветеринарное свидетельство, паспорт животного, список хозяев животных;

– создание статистических отчётов с возможностью выбора аналитики и оцениваемых параметров: о вакцинации животных, о наградах животных;

Существует много аналогов разрабатываемой программы, которые автоматизируют процесс регистрации животных и их владельцев для участия в выставках. Например, программа компании «World Expo», программа компании «Zoo show». Разрабатываемая программа является упрощённой версией этих программ.

## Проектирование модели

Цель моделирования данных состоит в обеспечении разработчика информационной системы концептуальной схемой базы данных в форме одной модели или нескольких локальных моделей, которые относительно легко могут быть отображены в любую систему баз данных.

Наиболее распространенным средством моделирования данных являются диаграммы «Сущность-связь» (ERD). С их помощью определяются важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи). ERD непосредственно используются для проектирования реляционных баз данных. Нотация ERD была впервые введена П. Ченном и получила дальнейшее развитие в работах Баркера.

Диаграмма «Сущность-связь» представлена на рисунке 1.5.

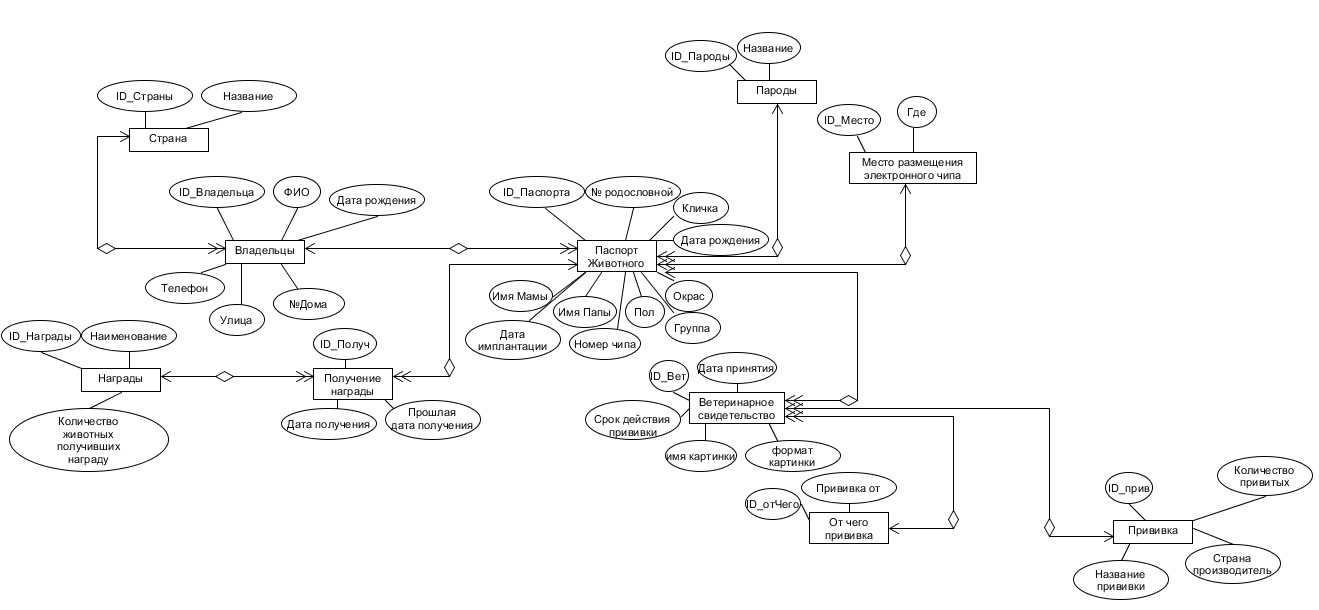


Рисунок 1.5

Исходя из исследования предметной области, можно выделить следующие сущности разработки: «Паспорт животного», «Пароды», «Место расположения электронного чипа», «Владельцы, Страна», «Награды», «Получение награды», «Ветеринарное свидетельство», «Отчего прививка», «Прививка».

Для сущности «Паспорт животного» можно выделить следующие атрибуты:

– № родословной;

– кличка;

– дата рождения;

– группа;

– окрас;

– пол;

– номер чипа;

– дата имплантации;

– имя папы;

– имя мамы.

Для сущности «Пароды» атрибутом будет являться «Название».

Для сущности «Место расположения электронного чипа» атрибутом будет являться «Где».

Для сущности «Владельцы» атрибутами является:

– фио;

– дата рождения;

– телефон;

– улица;

– № дома.

Для сущности «Страна» атрибутом будет являться «Название».

Для сущности «Награды» атрибутами являться:

– наименование,

– количество животных получивших награду.

Для сущности «Получение награды» атрибутами является:

– дата получения;

– прошлая дата получения.

Для сущности «Ветеринарное свидетельство» атрибутами является:

– дата принятия;

– срок действия прививки;

– имя картинки;

– формат картинки.

Для сущности «Отчего прививка» атрибутом будет являться «Прививка от».

Для сущности «Прививка» атрибутами является:

– название прививки;

– страна производитель,

– количество привитых.

Перечень этих диаграмм представляет собой неотъемлемую часть графической нотации унифицированного языка моделирования (UML), сам процесс объектно-ориентированного программирования (ООП) неразрывно связан с процессом построения этих диаграмм. Совокупность построенных таким образом диаграмм содержит всю информацию, необходимую для реализации проекта сложной системы.

В рамках языка UML все представления о модели сложной системы фиксируются в виде специальных графических конструкций – диаграмм. В терминах языка UML определены следующие виды диаграмм: диаграмма вариантов использования, диаграмма классов, диаграмма деятельности, диаграмма последовательности, диаграмма компонентов.

Суть диаграммы вариантов использования состоит в том, что проектируемая система представляется в виде множества сущностей или актёров, взаимодействующих с системой с помощью, так называемых, вариантов использования.

Суть диаграммы вариантов использования состоит в следующем: проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью, так называемых вариантов использования.

Варианты использования описывают не только взаимодействия между пользователями и сущностью, но также реакции сущности на получение отдельных сообщений от пользователей и восприятие этих сообщений за пределами сущности. Варианты использования могут включать в себя описание особенностей способов реализации сервиса и различных исключительных ситуаций, таких как корректная обработка ошибок системы. Множество вариантов использования в целом должно определять все возможные стороны ожидаемого поведения системы.

Пользователь представляет собой внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует её функциональные возможности для достижения определённых целей или решения частных задач. При этом актёры служат для обозначения согласованного множества ролей, которые могут играть пользователи в процессе взаимодействия с проектируемой системой. Каждый актёр может рассматриваться как некоторая отдельная роль относительно конкретного варианта использования. Суть диаграммы вариантов использования состоит в том, что проектируемая система представляется в виде множества сущностей или актёров, взаимодействующих с системой с помощью, так называемых, вариантов использования.

Данная программа имеет следующие основные функции:

– осуществление ведение базы данных;

– осуществление работы с базой данных;

– хранение графической информации;

– контроль календаря прививок;

– осуществление расчёта статистики;

– создание отчётов.

К вспомогательным функциям, расширяющим возможности системы относятся следующие функции:

– предоставление информации о странах;

– предоставление информации о паспортах животных;

– предоставление информации о ветеринарных свидетельствах;

– предоставление информации о владельцах;

– предоставление информации о наградах;

– предоставление информацию о полученных наградах;

– предоставление информацию о предназначенных прививках;

– предоставление информацию о породах;

– предоставление информацию о прививках;

– предоставление информацию о местах размещения электронного чипа;

– предоставление справочной информации.

Диаграмма вариантов использования представлена в графической части на листе 1.

Диаграмма классов служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма классов может отражать, в частности, различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывает их внутреннюю структуру и типы отношений. На данной диаграмме не указывается информация о временных аспектах функционирования системы. С этой точки зрения диаграмма классов является дальнейшим развитием концептуальной модели проектируемой системы. Диаграмма классов для проектируемой системы представлена в графической части на листе 2.

Полное описание некоторых основных классов проекта, которое состоит из трёх рахделов, представлено в таблице 1.1.

Таблица 1.1 – Состав некоторых основных классов проекта

|  |  |  |
| --- | --- | --- |
| Имя класса | Имя поля | Методы |
| aktyalizat | Aktyaliz, AnimalBDDataSet, components, DGV\_ветеринарное\_свидетельство, DGV\_Владельцы\_актул, DGV\_место\_размещ\_актул, DGV\_Награды\_актул, DGV\_отчего\_актул, DGV\_паспорт\_животного\_актул, DGV\_получ\_награды\_актул, DGV\_породы\_актул,  DGV\_прививки\_актул,  DGV\_Страна\_актул, GrBox1\_add-7 GrBox1\_del -7 GrBox1\_update-7, GroupBox\_add -7, GroupBox\_del-7, GroupBox\_update-7,  grup\_del, grup\_dobavit,  grup\_update,  id\_вет\_св\_дел,  id\_вет\_св\_доб,  id\_вет\_св\_обнв,  id\_владельца\_доб, id\_владельца\_животного\_доб,  …  ПородыBindingSource ПородыTableAdapter прививка\_от\_дел прививка\_от\_доб прививка\_от\_обнв ПрививкаBindingSource ПрививкаTableAdapter ПрививкаотDataGridViewTextBoxColumn прошлая\_дата\_получ\_доб прошлая\_дата\_получ\_обнв срок\_дейсвия\_вет\_доб срок\_дейсвия\_вет\_обнв страна\_производитель\_доб страна\_производитель\_обнв | aktyalizat\_KeyDown aktyalizat\_Load DGV\_Владельцы\_актул\_CellContentClick  Getdatalist 1-9 id\_страны\_доб\_KeyPress Pasport\_Click Poluch\_Nagrad\_Click save\_Click Vixod\_Click Vladelti\_Click Добавить\_вет\_свед\_Click Добавить\_место\_размещ\_чипа\_Click Добавить\_награду\_Click Добавить\_отчего\_прививка\_Click Добавить\_паспорт\_Click Добавить\_получ\_награды\_Click Добавить\_породу\_Click Добавить\_прививку\_Click Добавить\_страну\_Click Добавть\_владельца\_Click Изменить\_вет\_свед\_Click Изменить\_владельца\_Click Изменить\_место\_размещ\_чипа\_Click Изменить\_награду\_Click Изменить\_отчего\_прививка\_Click Изменить\_паспорт\_Click Изменить\_получ\_награды\_Click Изменить\_породу\_Click Изменить\_прививку\_Click Изменить\_страну\_Click название\_страны\_доб\_KeyPress Удалить\_вет\_свед\_Click Удалить\_владельца\_Click Удалить\_место\_размещ\_чипа\_Click Удалить\_награду\_Click Удалить\_отчего\_прививка\_Click Удалить\_паспорт\_Click Удалить\_получ\_награды\_Click Удалить\_породу\_Click |

Продолжение таблицы 1.1

|  |  |  |
| --- | --- | --- |
| Имя класса | Имя поля | Методы |
| prosmotr | Components DataGridView1-2 DGV\_ветеринарное\_свидетельство DGV\_Владельцы\_просмотр DGV\_Награды DGV\_паспорт\_животного DTP\_D1-2 DTP\_вет\_дата\_принятия1-2 DTP\_вет\_срок1-2 DTP\_паспорт\_ДатаРОЖД\_КОНЕЦ DTP\_паспорт\_ДатаРОЖД\_НАЧАЛО DTP\_паспорт\_имплант1-2 DTP\_получ1-2 DTP\_прошлое1-2 GroupBox1 GroupBox1-10 Label1-17 misValue report\_button1-4 report1-2 TabPage1-5 TextBox1-2 xlApp xlWorkBook xlWorkSheet закрыть Картинка параметр1поиска паспорт\_дата\_имплант\_поиск паспорт\_дата\_рожд\_поиск поиск\_владельцев поиск\_дата\_прин поиск\_по\_кличкам\_животного поиск\_по\_кличкам\_животных\_вет\_свид поиск\_по\_периоду Поиск\_получ поиск\_прош поиск\_срок ПоискВладельцев принять\_поиск\_по\_кличкам\_награды строка\_поиска\_по\_кличкам\_вет\_свид строка\_поиска\_по\_кличке | Button1\_Click DGV\_ветеринарное\_свидетельство\_SelectionChanged Getdatalist-2 prosmotr\_KeyDown prosmotr\_Load ReleaseObject report\_Button2\_Click -4 report1\_Click -2 Search\_nag Search\_Pass Search\_Vet Search\_Vlad закрыть\_Click паспорт\_дата\_имплант\_поиск\_Click паспорт\_дата\_рожд\_поиск\_Click поиск\_дата\_прин\_Click поиск\_по\_кличкам\_животного\_Click поиск\_по\_кличкам\_животных\_вет\_свид\_Click Поиск\_получ\_Click поиск\_прош\_Click поиск\_срок\_Click ПоискВладельцев\_Click  принять\_поиск\_по\_кличкам\_награды\_Click |
| StartForm | Components Exit\_app MainMenu Read\_INFO Spravka update\_INFO | Exit\_app\_Click Read\_INFO\_Click Spravka\_Click StartForm\_KeyDown StartForm\_Load update\_INFO\_Click |

При моделировании поведения проектируемой или анализируемой системы возникает необходимость детализировать особенности алгоритмической и логической реализации выполняемых системой операций. Для моделирования процесса выполнения операций в языке UML используются так называемые диаграммы деятельности. Каждое состояние на диаграмме деятельности соответствует выполнению некоторой элементарной операции, переход в следующее состояние срабатывает только при завершении этой операции. Графически диаграмма деятельности представляется в форме графа, вершинами которого являются состояния действия, а дугами - переходы от одного состояния действия к другому. Основная цель использования диаграмм деятельности - визуализация особенностей реализации операций классов, когда необходимо представить алгоритмы их выполнения. Диаграмма деятельности представлена в графической части на листе 3.

Для моделирования взаимодействия объектов в UML используются соответствующие диаграммы взаимодействия. Если рассматривать взаимодействия объектов во времени, тогда для представления временных особенностей передачи и приема сообщений между объектами используется диаграмма последовательности.

Временной аспект поведения имеет существенное значение при моделировании синхронных процессов, описывающих взаимодействия объектов. Именно для этой цели и используются диаграммы последовательности, в которых ключевым моментом является динамика взаимодействия объектов во времени. При этом диаграмма последовательности имеет как бы два измерения: одно - слева направо в виде вертикальных линий, каждая из которых изображает линию жизни отдельного объекта, участвующего во взаимодействии; второе - вертикальная временная ось, направленная сверху вниз, на которой начальному моменту времени соответствует самая верхняя часть диаграммы. Диаграмма последовательности для проектируемой системы представлена в графической части на листе 4.

Рассмотренные ранее диаграммы отражали концептуальные аспекты построения модели системы и относились к логическому уровню представления. Особенность логического представления заключается в том, что оно оперирует понятиями, которые не имеют самостоятельного материального воплощения. Другими словами, различные элементы логического представления, такие как классы, ассоциации, состояния, сообщения, не существуют материально или физически. Они лишь отражают наше понимание структуры физической системы или аспекты ее поведения.

Основное назначение логического представления состоит в анализе структурных и функциональных отношений между элементами модели системы. Однако для создания конкретной физической системы необходимо некоторым образом реализовать все элементы логического представления в конкретные материальные сущности. Для описания таких реальных сущностей предназначен другой аспект модельного представления, а именно физическое представление модели.

Диаграмма компонентов описывает объекты реального мира - компоненты программного обеспечения. Эта диаграмма позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами. Вид диаграммы компонентов для данной проектируемой системы представлен в графической части на листе 5. Она содержит следующие компоненты:

– программные компоненты, созданные в среде Visual Studio: «.vb», «.resx»;

– файл базы данных «animalBD.mdf»;

– журнал транзакций «animalBD\_log.ldf»;

– отчёты «\*.xlsx»;

– файл справочной системы «ZelZooExpo\_Info.chm».

# Вычислительная система

## 2.1 Требования к аппаратным и операционным ресурсам

Конфигурация компьютера, на котором будет разрабатываться программное приложение:

– процессор Intel Core i5 7600 3,50 Ггц;

– оперативная память DDR4 8Гбайт;

– жёсткий диск HDD Toshiba 2Тбайт;

– видеокарта AMD Radeon R7 370 4Гбайт;

– материнская плата Asus;

– монитор Viewsonic 24 дюйма.

Для удобной работы с программой необходимо наличие мыши и клавиатуры.

## Инструменты разработки

Инструментами разработки программного средства будут являться:

– операционная система Windows 10;

– среда разработки Microsoft Visual Studio 2017;

– язык программирования Visual Basic.net;

– система управления базами данных Microsoft SQL Server 2017;

– программа для создания справки Help & Manual;

– программа для создания диаграмм UMLet;

– офисный пакет MS Office;

* среда разработки базы данных sql server management studio;

– программа для создания инсталлятора Smart Install Maker.

Windows 10 – [операционная система](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) для [персональных компьютеров](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%81%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80) и [рабочих станций](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B1%D0%BE%D1%87%D0%B0%D1%8F_%D1%81%D1%82%D0%B0%D0%BD%D1%86%D0%B8%D1%8F), разработанная корпорацией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) в рамках семейства [Windows](https://ru.wikipedia.org/wiki/Windows_NT) . После [Windows 8.1](https://ru.wikipedia.org/wiki/Windows_8.1) система получила номер 10, минуя 9. Серверный аналог Windows 10 – [Windows Server 2016](https://ru.wikipedia.org/wiki/Windows_Server_2016). Система призвана стать единой для разных устройств, таких как [персональные компьютеры](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%81%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80), [планшеты](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D0%BD%D1%88%D0%B5%D1%82%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80), [смартфоны](https://ru.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82%D1%84%D0%BE%D0%BD), консоли

[Xbox One](https://ru.wikipedia.org/wiki/Xbox_One) [6].

Microsoft Visual Studio 2017 – средство разработки [программного обеспечения](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), разрабатываемое корпорацией [Microsoft](http://ru.wikipedia.org/wiki/Microsoft) и включающее [язык программирования](http://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) и [среду разработки](http://ru.wikipedia.org/wiki/%D0%A1%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8). В то же время Visual Basic 2017 сочетает в себе [процедуры](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%D1%80%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) и элементы [объектно-ориентированных](http://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) и [компонентно-ориентированных](http://ru.wikipedia.org/w/index.php?title=%D0%9A%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F&action=edit&redlink=1) языков программирования. Среда разработки Visual Basic 2017 включает инструменты для визуального конструирования пользовательского интерфейса [9].

Язык программирования Visual Basic (VB) включает инструменты для визуального конструирования пользовательского интерфейс. Visual Basic считается хорошим [средством быстрой разработки](http://ru.wikipedia.org/wiki/RAD_%28%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%29) прототипов программы, для разработки приложений баз данных и вообще для компонентного способа создания программ, работающих под управлением операционных систем семейства [Microsoft Windows](http://ru.wikipedia.org/wiki/Microsoft_Windows) [10].

Microsoft SQL Server 2017 – система управления реляционными базами данных (СУРБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими Система управления базами данных (СУБД) в этом сегменте рынка [8].

Help & Manual – это приложение для быстрого создания файлов справки (help-файлов), справочных систем, on-line руководств пользователя, пособий и технической документации к программному обеспечению и техническим системам. Help & Manual 7 поддерживает вывод документации в такие форматы, как hypertext markup language (HTML) (on-line руководства), chemical tanker (CHM) (файлы справки для операционной системы (ОС) MS Windows), rich text format (RTF), и portable document format (PDF). И все это на основе единого проекта-источника [11].

UMLet – это [инструмент UML](https://en.wikipedia.org/wiki/UML_tool) с [открытым исходным кодом на](https://en.wikipedia.org/wiki/Open-source) основе [Java](https://en.wikipedia.org/wiki/Java_(programming_language))[,](https://en.wikipedia.org/wiki/UML_tool)предназначенный для обучения [унифицированному языку моделирования](https://en.wikipedia.org/wiki/Unified_Modeling_Language)и для быстрого создания диаграмм UML. Это инструмент рисования, а не инструмент моделирования, поскольку нет базового словаря или каталога объектов многократного использования. UMLet распространяется по [лицензии GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License) . UMLet имеет простой пользовательский интерфейс, который использует коды форматирования текста для изменения основных фигур с помощью украшений и аннотаций, поэтому нет леса икон или диалогов списка параметров по-своему. Это требует от пользователя изучения еще одного языка разметки текста, но усилия небольшие, а разметка очевидна для опытного дизайнера [12].

[Microsoft Office](https://ru.wikipedia.org/wiki/Microsoft_Word)  –  [офисный пакет](https://ru.wikipedia.org/wiki/%D0%9E%D1%84%D0%B8%D1%81%D0%BD%D1%8B%D0%B9_%D0%BF%D0%B0%D0%BA%D0%B5%D1%82) [приложений](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), созданных корпорацией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) для операционных [Microsoft Windows](https://ru.wikipedia.org/wiki/Microsoft_Windows), [Windows Phone](https://ru.wikipedia.org/wiki/Windows_Phone), [Android](https://ru.wikipedia.org/wiki/Android), [OS X](https://ru.wikipedia.org/wiki/OS_X), [iOS](https://ru.wikipedia.org/wiki/IOS). В состав этого пакета входит программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др. Microsoft Office является сервером [OLE](https://ru.wikipedia.org/wiki/Object_Linking_and_Embedding)-объектов и его функции могут использоваться другими приложениями, а также самими приложениями Microsoft Office. Поддерживает скрипты и макросы, написанные на [VBA](https://ru.wikipedia.org/wiki/Visual_Basic_for_Applications). [7].

SQL Server Management Studio (SSMS) — утилита из [Microsoft SQL Server](https://ru.wikipedia.org/wiki/Microsoft_SQL_Server) 2005 и более поздних версий для конфигурирования, управления и [администрирования](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D0%BE%D0%B5_%D0%B0%D0%B4%D0%BC%D0%B8%D0%BD%D0%B8%D1%81%D1%82%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) всех компонентов Microsoft SQL Server. Утилита включает [скриптовый](https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D1%80%D0%B8%D0%BF%D1%82) редактор и графическую программу, которая работает с объектами и настройками сервера.Главным инструментом SQL Server Management Studio является Object Explorer, который позволяет пользователю просматривать, извлекать объекты сервера, а также полностью ими управлять.Также есть SQL Server Management Studio Express для Express версии сервера, которая является бесплатной. Однако в ней нет поддержки ряда компонентов (Analysis Services, Integration Services, Notification Services, Reporting Services) и SQL Server 2005 Mobile Edition [14].

Smart Install Maker – утилита предоставляет алгоритм Cabinet для сжатия и создания компактных инсталляторов, поддерживает создание многоязычного инсталлятора (до 25 языков), а также предоставляет средства для управления внешним видом программы-установки [13].

# 3 Проектирование задачи

## 3.1 Требования к приложению

Данное приложение не требует никаких специальных средств защиты либо ограничений прав доступа к данным.

Разрабатываемое приложение должно иметь понятный и удобный в использовании интерфейс, чтобы взаимодействовать между программой и пользователям было максимально упрощённо. Добавление, удаление и изменение будет происходить с помощью ленточной формы. Для более удобного перемещение между таблицами будут использованы вкладки. Программное приложение должно обеспечить оперативный поиск необходимой информации по различным критериям, для этого будут разработаны процедуры обработки информации.

Все входные данные должны проверяться на ошибки. При совершении пользователям ошибки, ему будет предоставлено диалоговое окно с разъяснением ошибки.

Кроме этого, формы в приложении должны быть выполнены в едином стиле, сдержанной цветовой гамме, иметь стандартные элементы управления данными, не должны быть перегружены информацией, и в свое время понятны простому пользователю.

При конструировании форм в необходимых случаях нужно предусмотреть возможность защиты данных от изменений, например при просмотре информации, установить ограничения на корректировку записей, а также отсутствие данных при выполнении отбора записей в базе данных.

В создаваемом программном средстве должна быть справочная система, в которой будут предусмотрены все возможные способы работы системы. Справочная система должна содержать ответы на вопросы, которые могут возникнуть у пользователя.

## 3.2 Концептуальный прототип

Концептуальный прототип состоит из описания внешнего пользовательского интерфейса, а именно, элементов управления.

При создании данного приложения важную роль играют формы, так как они являются основным диалоговым средством работы пользователя. Разрабатываемое приложение будет содержать главную форму и две дополнительных. Такая структура интерфейса позволит классифицировать основные функции программы по группам.

При проектировании концептуального прототипа предполагается, что при запуске программы первой будет загружаться форма «Главное меню», макет которой представлен на рисунке 3.1.



Рисунок 3.1

На главной форме располагаются кнопки «Актуализация данных», «Просмотр данных», которые осуществляют переход на соответствующие формы. Кнопки «Справка», «Выход» предназначены для открытия справочной системы и выхода из программы соответственно. Переходы на другие формы осуществляется с помощью компонента «Button».

На форме «Актуализация данных» располагаются текстовые поля для ввода данных, таблицы для просмотра информации, кнопки «Добавить», «Удалить», «Изменить», «Сохранить» для работы с данными и кнопка «Назад» для возврата на главную форму. Макет формы «Актуализация данных» представлен на рисунке 3.2.

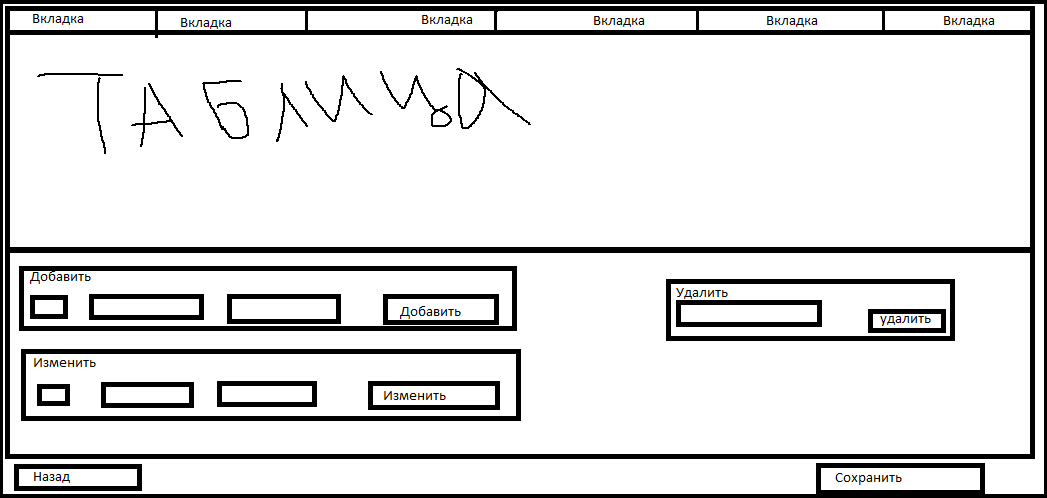


Рисунок 3.2

На форме «Актуализация данных» присутствует система вкладок для работы с другими таблицами соответственно: «Страна», «Награды», «Отчего прививка», «Прививки», «Место размещения электронного чипа», «Породы», «Владельцы», «Паспорт животного», «Получение награды», «Ветеринарное свидетельство». Макеты форм описанных вкладок аналогичные, представленному на рисунке 3.2. Для перехода по вкладкам используется компонент «Tab Control», для отображения данных из базы данных, используется компонент «Data Grid View», для ввода данных используется компонент Text Box, для добавления, удаления и изменения используются компоненты «Button».

На форме «Просмотр данных» располагаются текстовые поля для сортировки и поиска данных, таблицы для просмотра информации, кнопки «Сортировать / поиск», «Создать отчёт» и кнопка «Назад» для возврата на главную форму. Макет формы «Просмотр данных» представлен на рисунке 3.3.

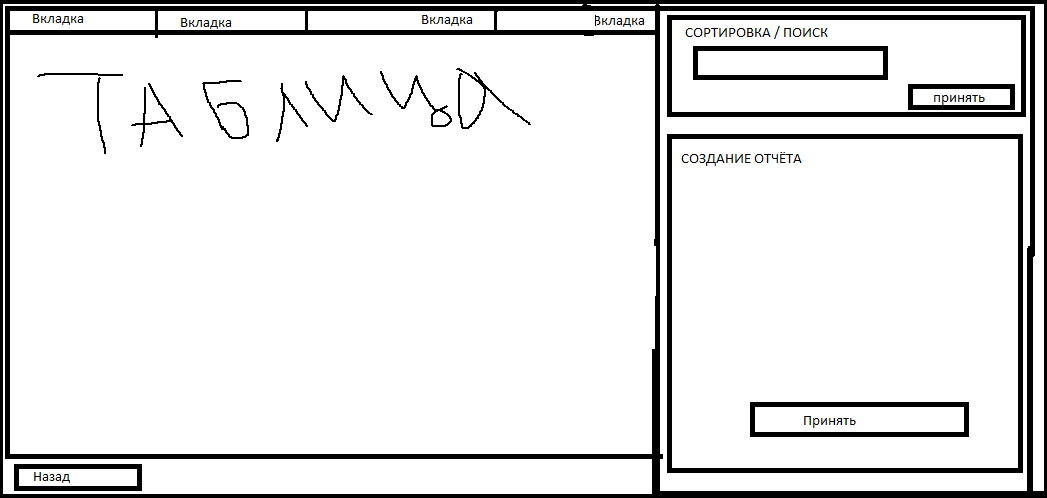


Рисунок 3.3

На форме «Просмотр данных» присутствует система вкладок для работы с другими таблицами соответственно: «Владельцы», «Паспорт животного», «Ветеринарное свидетельство», «Награды». Макеты форм соответствующих вкладок аналогичные, представленному на рисунке 3.3. Для перехода по вкладкам используется компонент «Tab Control», для отображение данных из таблицы базы данных используется компонент «Data Grid View», для ввода данных используется компонент «Text Box», для сортировки, поиска, создания отчётов используются компоненты «Button».

## 3.3 Организация данных

Организация данных подразумевает создание модели данных, главными элементами которой являются сущности и их связи.

Реляционная модель основана на математическом понятии отношения, представлением которого является таблица. В реляционной модели отношения используются для хранения информации об объектах, представленных в базе данных. Отношение имеет вид двухмерной таблицы, в которой строки соответствуют записям, а столбцы - атрибутам. Каждая запись должна однозначно характеризоваться в таблице. Для этого используют первичные и вторичные ключи. Достоинством реляционной модели является простота и удобство физической реализации.

Реляционная модель базы данных подразумевает нормализацию всех таблиц данных. Нормализация – это формальный метод анализа отношений на основе их первичного ключа и функциональных зависимостей, существующих между их атрибутами.

Структура базы данных разрабатываемого программного средства включает 10 таблиц.

Структура данных таблиц, и их краткое описание приводится в таблицах 1-10.

Таблица «Страна» хранит информацию о странах. Структура приведена в таблице 3.1.

Таблица 3.1- Структура таблицы «Страна»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Страны | int | 4 | Идентификатор страны |
| Название | varchar | 20 | Наименование страны |

Таблица «Награды» хранит информацию о наградах. Структура приведена в таблице 3.2.

Таблица 3.2- Структура таблицы «Награды»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Награды | int | 4 | Идентификатор награды |
| Наименование | varchar | 100 | Наименование награды |
| Количество\_животных\_получивших\_награду | int | 4 | Количество животных получивших награду |

Таблица «ОтчегоПрививка» хранит информацию о предназначенных прививках. Структура приведена в таблице 3.3.

Таблица 3.3- Структура таблицы «ОтчегоПрививка»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_отЧего | int | 4 | Идентификатор назначения прививки |
| Прививка\_от | varchar | 20 | Назначения прививки |

Таблица «Прививка» хранит информацию о прививках. Структура приведена в таблице 3.4.

Таблица 3.4- Структура таблицы «Прививка»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_прив | int | 4 | Идентификатор прививки |
| Название\_прививки | varchar | 20 | Наименование прививки |
| Страна\_производитель | varchar | 20 | Страна производитель прививки |
| Количество\_привитых | int | 4 | Количество привитых |

Таблица «Место\_размещения\_электронного\_чипа» хранит информацию о месте размещения электронного чипа. Структура приведена в таблице 3.5.

Таблица 3.5- Структура таблицы «Место\_размещения\_электронного\_чипа»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_места | int | 4 | Идентификатор чипа |
| Где | varchar | 20 | Место расположения чипа |

Таблица «Породы» хранит информацию о породах. Структура приведена в таблице 3.6.

Таблица 3.6- Структура таблицы «Породы»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_породы | int | 4 | Идентификатор породы |
| Название | varchar | 20 | Наименование породы |

Таблица «Владельцы» хранит информацию о владельцах. Структура приведена в таблице 3.7.

Таблица 3.7- Структура таблицы «Владельцы»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Владельца | int | 4 | Номер владельца |
| ФИО | varchar | 30 | Фамилия, имя, отчество владельца |
| Дата\_рождения | date | 3 | Дата рождения владельца |
| Страна | int | 4 | Идентификатор таблицы «Страна» |
| Город | varchar | 20 | Город владельца |
| Улица | varchar | 20 | Улица владельца |
| Номер\_дома | int | 4 | Номер дома где проживает владелец |

Таблица «Паспорт\_животного» хранит информацию о паспортах животных. Структура приведена в таблице 3.8.

Таблица 3.8- Структура таблицы «Паспорт\_животного»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Животного | int | 4 | Идентификатор животного |
| Владелец | int | 4 | Идентификатор таблицы «Владельцы» |
| Номер\_родословной | int | 4 | Номер родословной |
| Кличка | varchar | 20 | Кличка животного |
| Порода | int | 4 | Идентификатор таблицы «Породы» |
| Дата\_рождения | date | 3 | Дата рождения животного |
| Группа | varchar | 20 | Группа животного (кошка или собака) |
| Окрас | varchar | 20 | Окрас животного |
| Пол | varchar | 1 | Пол животного |
| Имя\_мамы | varchar | 20 | Имя мамы животного |
| Имя\_папы | varchar | 20 | Имя папы животного |
| Номер\_чипа | int | 4 | Идентификатор чипа животного |
| Дата\_имплантации | date | 3 | Дата имплантации электронного чипа |
| Место\_размещения | int | 4 | Идентификатор таблицы «Место\_размещения\_электронного\_чипа» |

Таблица «Получение\_награды» хранит информацию о полученных наградах. Структура приведена в таблице 3.9.

Таблица 3.9- Структура таблицы «Получение\_награды»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Получ | int | 4 | Идентификатор получения награды |
| Кличка\_животного | int | 4 | Идентификатор таблицы «Паспорт животного» |
| Наименование\_награды | int | 4 | Идентификатор таблицы «Награды» |
| Дата\_получения | date | 3 | Дата получения награды |
| Прошлая\_дата\_получения | date | 3 | Прошлая дата получения награды |

Таблица «Ветеринарное\_свидетельство» хранит информацию о ветеринарном свидетельстве животных. Структура приведена в таблице 3.10.

Таблица 3.10- Структура таблицы «Ветеринарное\_свидетельство»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля, байт | Описание поля |
| ID\_Вет | int | 4 | Идентификатор ветеринарного свидетельства |
| Имя\_животного | int | 4 | Идентификатор таблицы «Паспорт животного» |
| Наименование\_прививки | int | 4 | Идентификатор таблицы «Прививки» |
| Прививка\_от | int | 4 | Идентификатор таблицы «ОтчегоПрививка» |
| Дата\_приниятия | date | 3 | Дата принятия прививки |
| Срок\_действия\_прививки | date | 3 | Срок действия прививки |
| Имя\_картинки | varchar | 20 | Имя картинки |
| Формат\_картинки | varchar | 4 | Формат картинки |

Физическая структура базы данных представлена на рисунке 3.4

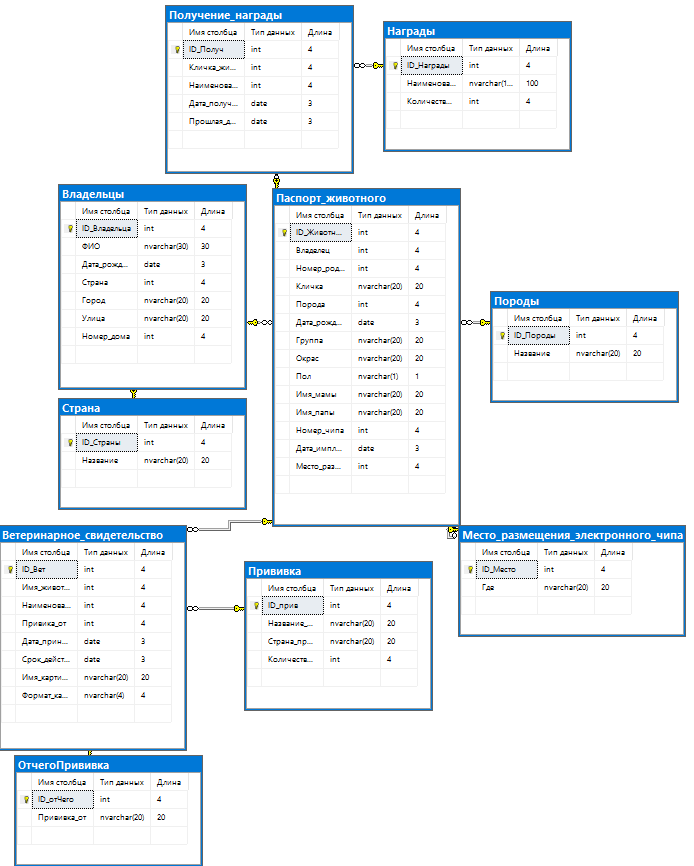


Рисунок 3.4

## 3.4 Функции и элементы управления

Основными функциями данного проекта являются подсчёт статистической информации.

При добавлении данных в таблицу «Ветеринарное свидетельство» будет срабатывать триггер, который подсчитывает количество привитых животных. Код триггера представлен ниже.

Create trigger trigger\_add\_info\_PRIV

on Ветеринарное\_свидетельство

for insert

as

update Прививка

set Количество\_привитых = s.Количество\_привитых +1

from Прививка s join

inserted i

on s.ID\_прив = i.Наименование\_прививки

При удалении данных из таблицы «Ветеринарное свидетельство» будет срабатывать триггер, который пересчитывает количество привитых животных. Код триггера представлен ниже.

Create trigger trigger\_del\_info\_PRIV

on Ветеринарное\_свидетельство

for delete

as

update Прививка

set Количество\_привитых = s.Количество\_привитых -1

from Прививка s join

deleted d

on s.ID\_прив = d.Наименование\_прививки

При добавлении данных в таблицу «Получение награды» будет срабатывать триггер, который подсчитывает количество животных, которые получили награду. Код триггера представлен ниже.

Create trigger trigger\_add\_info\_NAGRAD

on Получение\_награды

for insert

as

update Награды

set Количество\_животных\_получивших\_награду = s.Количество\_животных\_получивших\_награду +1

from Награды s join

inserted i

on s.ID\_Награды = i.Наименование\_награды

При удалении данных из таблицы «Получение награды» будет срабатывать триггер, который пересчитывает количество животных, которые получили награду. Код триггера представлен ниже.

Create trigger trigger\_del\_info\_NAGRAD

on Получение\_награды

for delete

as

update Награды

set Количество\_животных\_получивших\_награду = s.Количество\_животных\_получивших\_награду -1

from Награды s join

deleted d

on s.ID\_Награды = d.Наименование\_награды

Полный текст программы представлен в приложении А.

## 3.5 Проектирование справочной системы

Для корректной работы с приложением требуется обеспечить пользователя справочной системой, в которой будут приведены приемы работы с приложением, включающие данные о том, что произойдёт после нажатия на определенную кнопку или при выборе вкладки.

Справочная система необходима для ознакомления с программой. В ней должна присутствовать информация, которая поможет в решении проблемы с приложением, а также может напомнить, как пользоваться программным средством.

Система справки данного приложения будет содержать следующие разделы:

– «Введение»;

– «Инструкция»;

– «Об авторе».

Справочная система будет создана с помощью программного средства Help & Manual.

Help & Manual – это приложение для быстрого создания файлов справки (help-файлов), справочных систем, on-line руководств пользователя, пособий и технической документации к программному обеспечению и техническим системам. Help & Manual 7 поддерживает вывод документации в такие форматы, как hypertext markup language (HTML) (on-line руководства), chemical tanker CHM (файлы справки для операционной системы (ОС) MS Windows), rich text format (RTF), и portable document format (PDF). И все это на основе единого проекта-источника [11].

Справка будет вызываться клавишей F1 на всех формах, а также при нажатии на кнопку «Справка», которая находиться на главном форме.

# 4 Описание программного средства

## 4.1 Общие сведения

Приложение «ZelZooExpo.exe» предназначено для учёта информации о прививках и наградах животных. Файл базы данных называется «animalBD.mdb» вес, которого составляет 8 Мбайт. Само приложение весит 80 Мбайт. Для успешного использования программы, на компьютере должно быть свободного места на диске не меньше 100 Мбайт, оперативная память должна быть от 100 Мбайт. Приложение использует стандартные библиотеки такие как:

– Imports System.Configuration;

– Imports System.Data;

– Imports System.Data.SqlClient;

– Imports Excel = Microsoft.Office.Interop.Excel;

– Imports Microsoft.Office;

– Imports Microsoft.Office.Interop;

– Imports System.IO;

– Imports System.Reflection;

– Imports Microsoft.Reporting;

– Imports Microsoft.ReportingServices;

Для запуска приложения всего лишь запустить ярлык «ZelZooExpo.exe».

Компьютер должен работать под управлением операционной системы, начиная с Windows 7 и выше.

Инсталляция производится с установочного диска.

## 4.2 Функциональное назначение

Назначение программного средство «ZelZooexpo» является автоматизации работы организации, которая занимается выставкой животных. В данной программе классом решаемых задач являются:

– возможность добавлять, изменять и удалять данные в таблицы;

– возможность сортировать, фильтровать и искать данные в различных таблицах;

– возможность создавать отчёты по различным критериям.

## 4.3 Входные и выходные данные

Входными является данные, вводимые пользователем в таблицы базы данных: «Страна», «Награды», «Отчего прививка», «Прививка», «Породы», «Место размещения электронного чипа», «Владельцы», «Ветеринарное свидетельство», «Получение награды», «Паспорт животного». Кроме того, входными являются данные вводимые пользователем в поля для поиска и фильтрации данных.

При добавлении новой информации в таблицу «Владельцы» необходимо ввести следующие данные:

– фио;

– дата рождения;

– страна;

– город;

– улица;

– номер дома.

При добавлении новой информации в таблицу «Ветеринарное свидетельство» необходимо ввести следующие данные:

– имя животного;

– наименование прививки;

– прививка от;

– дата принятия;

– срок действия;

– имя картинки;

– формат картинки.

При добавлении новой информации в таблицу «Паспорт животного» необходимо ввести следующие данные:

– владелец;

– номер родословной;

– кличка;

– порода;

– дата рождения;

– группа;

– окрас;

– пол;

– имя мамы;

– имя папы;

– номер чипа;

– дата имплантации;

– место размещения.

Выходная информация будет представлена в виде отчётов экспортируемых в файлы MS Office.

Сформированный отчёт, содержащий информации животных, представлен на рисунке 4.1.



Рисунок 4.1

Сформированный отчёт, содержащий информации о владельцах животных, представлен на рисунке 4.2.

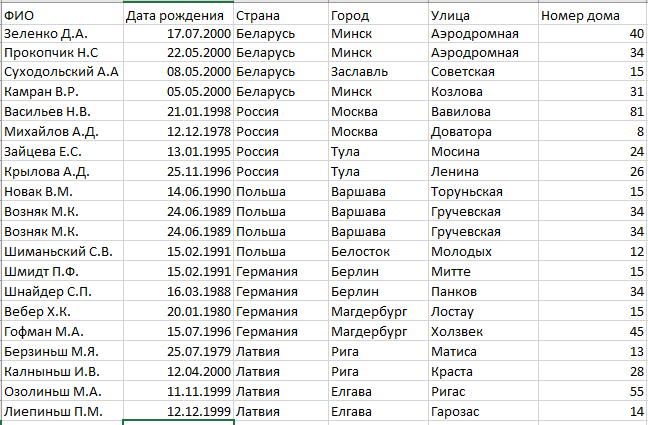


Рисунок 4.2

Сформированный отчёт, содержащий информации о прививках животных, представлен на рисунке 4.3.



Рисунок 4.3

Сформированный отчёт, содержащий информации о наградах животных, представлен на рисунке 4.4

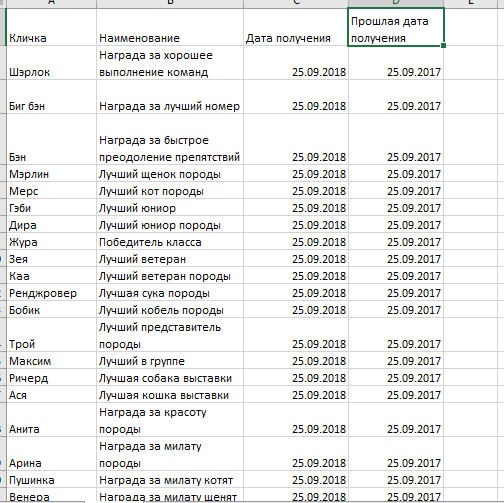


Рисунок 4.4

Сформированный отчёт, содержащий статистическую информацию о вакцинации животных, представлен на рисунке 4.5.

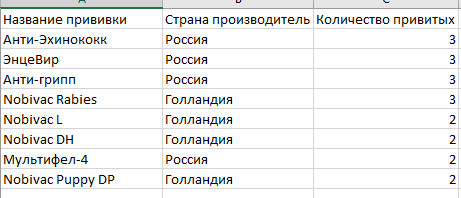


Рисунок 4.5

Сформированный отчёт, содержащий статистическую информацию о наградах животный, представлен на рисунке 4.6.

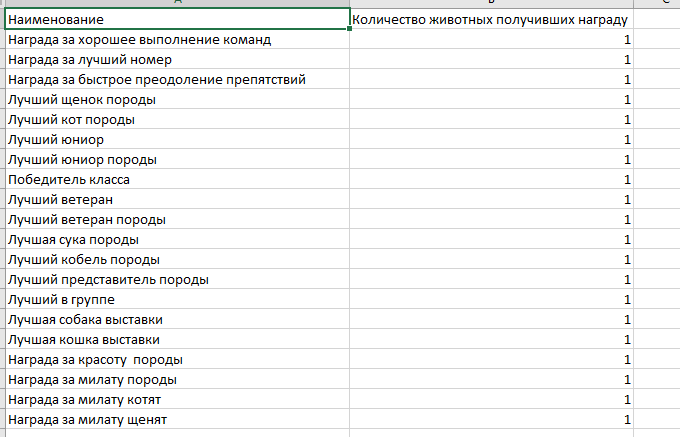


Рисунок 4.6

# 5 Методика испытаний

## 5.1 Технические требования

Минимальные системные требования к приложению:

* 20 Мбайт свободного места на винчестере;
* 100 Мбайт свободной оперативной памяти;
* процессор Intel Pentium 1.5 МГц или совместимый аналог;
* видеокарта с объемом памяти не менее 20 Мбайт.
* операционная система семейства Microsoft Windows .Net.

## 5.2 Функциональное тестирование

В процессе написания программного средства необходимо производить тестирование на правильность работы приложения. Одной из основных задач тестирования является устранение ошибок, происходящих при вводе данных.

Функциональное тестирование – это тестирование функций приложения на соответствие требованиям. Оценка производится в соответствии с ожидаемыми и полученными результатами (на основании функциональной спецификации), при условии, что функции отрабатывали на различных значениях.

Тестирование программы будет производиться последовательно, переходя из одной части программы в другую. Во время теста будут проверяться все действия с программой, навигация пунктам меню, которые может произвести пользователь. После чего, все собранные и найденные ошибки будут исправлены.

Таблица 5.1 – Test-case для добавления данных в базу данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 001 | Позитивный | Добавление данных в базу данных путём заполнения текстовых полей и нажатия на кнопку «Добавить» | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Актуализация сведений»;  3. Заполнить поля:  id Страны – 6  Название - США  4. Нажать кнопку «Добавить». | Ожидаемый результат  Данные добавлены в таблицу «Страна» |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат добавления данных представлена на рисунке 5.1. |

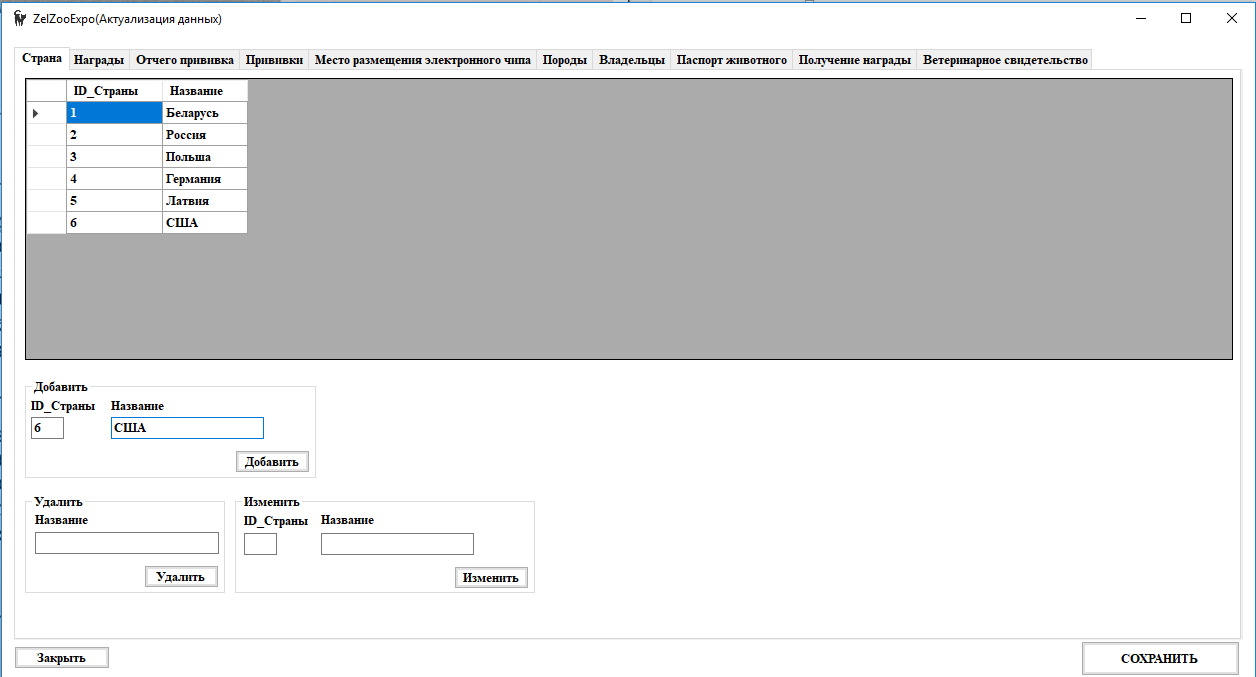


Рисунок 5.1

Таблица 5.2 – Test-case для изменения данных в базе данных.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 002 | Позитивный | Изменение данных в базе данных путём заполнения текстовых полей и нажатия на кнопку «Изменить» | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Актуализация сведений»;  3. Заполнить поля:  id Страны - 6  Название - ОАЭ  4. Нажать кнопку «Изменить». | Ожидаемый результат  Данные изменены в таблице «Страна» |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат изменения данных представлена на рисунке 5.2. |

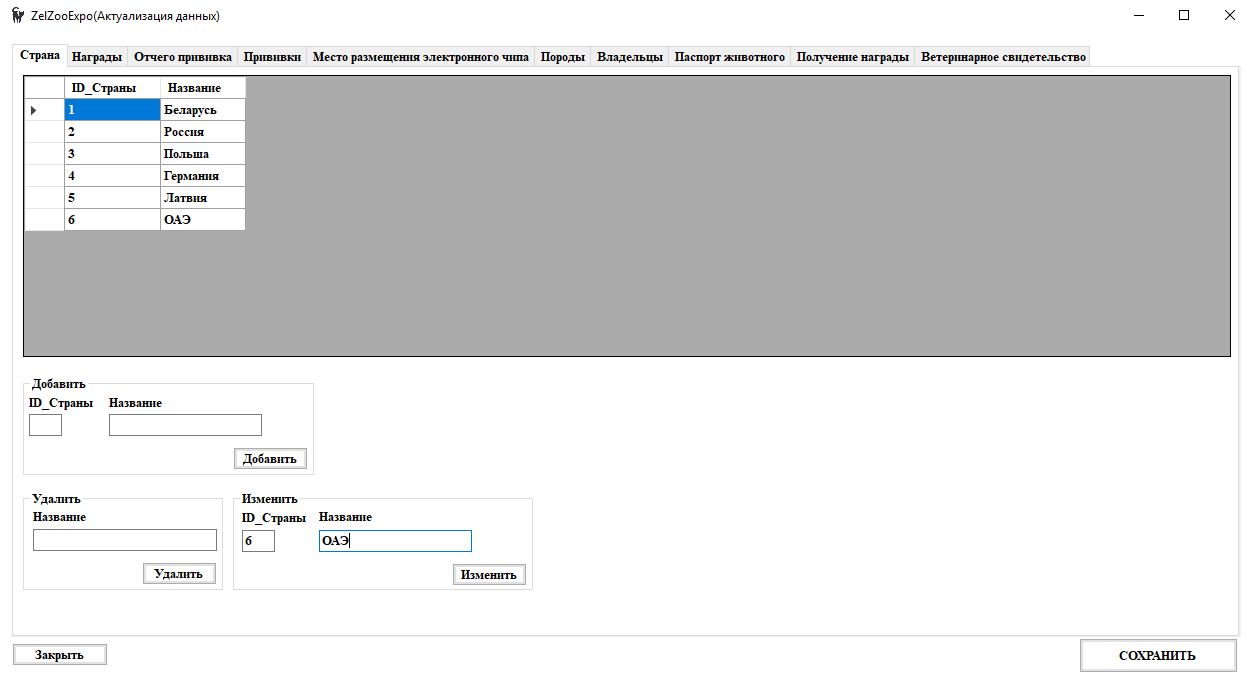


Рисунок 5.2

Таблица 5.3 – Test-case для удаления данных в базе данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 003 | Позитивный | Удаление данных в базе данных путём заполнения текстовых полей и нажатия на кнопку «Удалить» | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Актуализация сведений»;  3. Заполнить поле: Название - ОАЭ  4. Нажать кнопку «Удалить». | Ожидаемый результат  Данные удалены из таблицы «Страна» |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат удаления данных представлена на рисунке 5.3. |

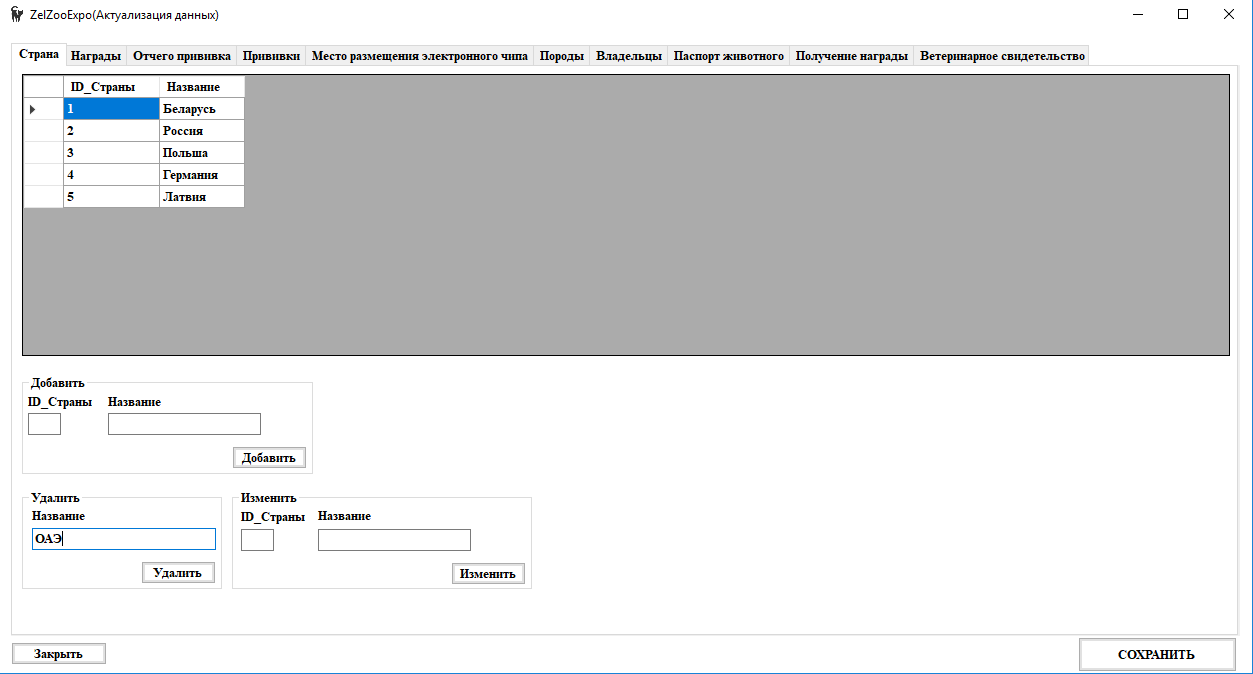


Рисунок 5.3

Таблица 5.4 – Test-case для сортировки данных в бузе данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 004 | Позитивный | Сортировка данных в базе данных путём нажимания на название столбца | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации»;  3. Нажать на название столбца. | Ожидаемый результат  При нажатии на название столбца данные отсортируются в таблице «Владельцы». |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат сортировки данных представлена на рисунке 5.4. |

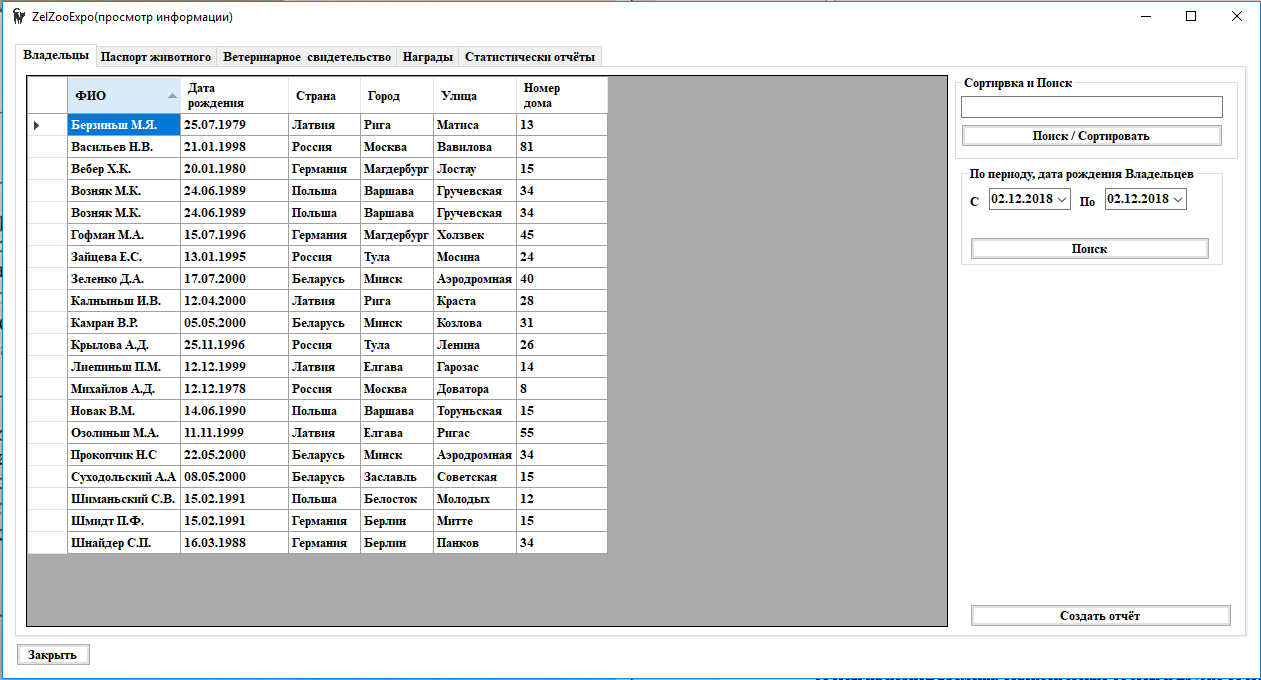


Рисунок 5.4

Таблица 5.5 – Test-case для фильтрации данных в бузе данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 005 | Позитивный | Фильтрация и сортировка данных в базе данных путём выбора периода | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации»;  3. Указать период;  4. Нажать на кнопку «Поиск». | Ожидаемый результат  Отображение данных соответствующих выбранному периоду. |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат фильтрации данных представлена на рисунке 5.5. |

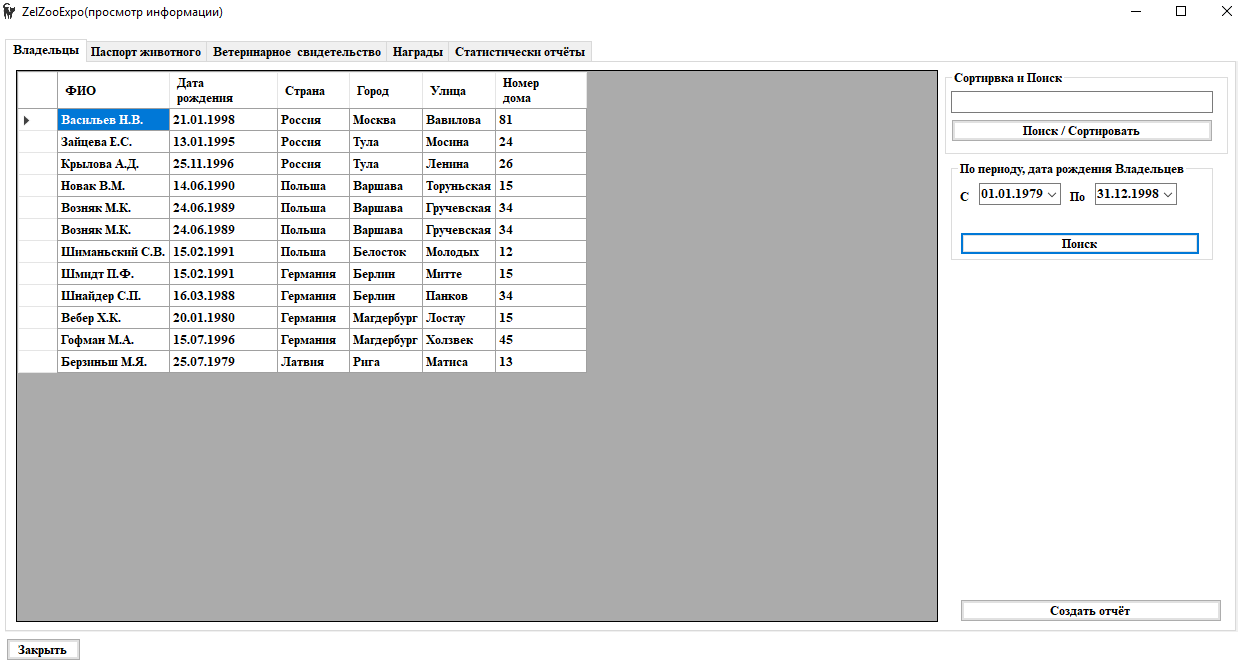


Рисунок 5.5

Таблица 5.6 – Test-case для поиска данных в базе данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 006 | Позитивный | Поиск данных в базе данных путём выбора периода | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации»;  3. Заполнить текстовое поле для поиска;  4. Нажать на кнопку «Поиск/Сортировка». | Ожидаемый результат  При нажатии на кнопку «Поиск/Сортировка» найдётся введённое значение. |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат поиска данных представлена на рисунке 5.6. |

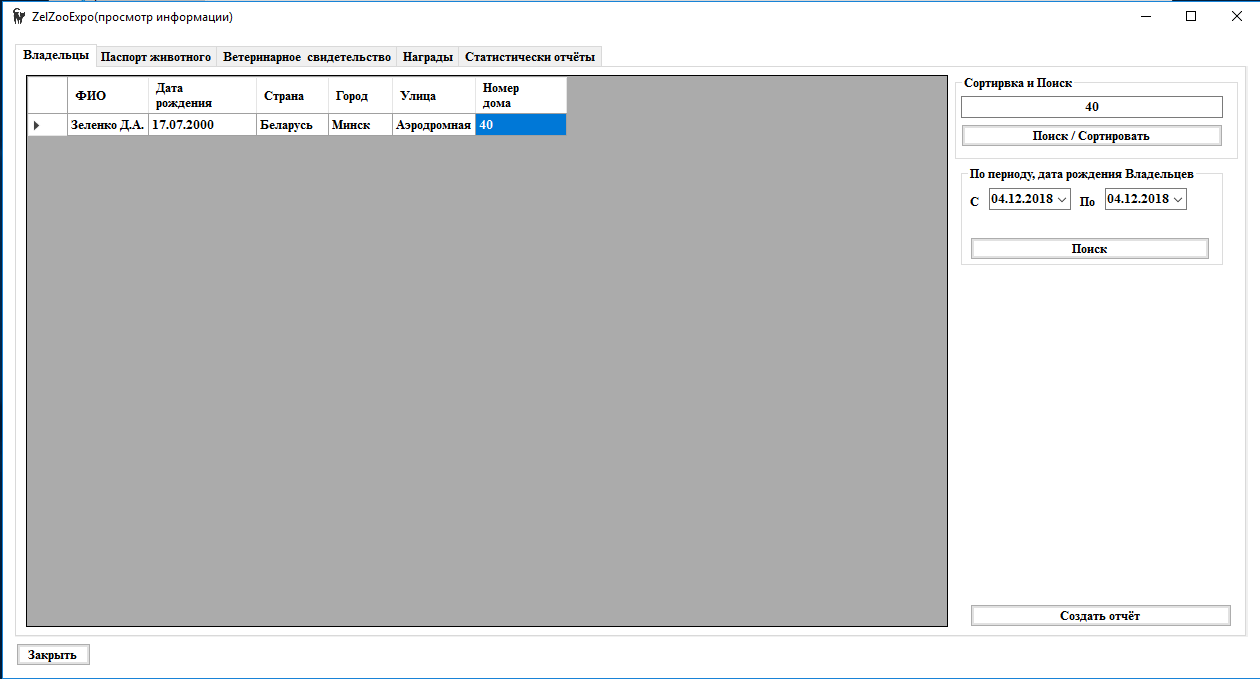


Рисунок 5.6

Таблица 5.7 – Test-case для вывода графической информации

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 007 | Позитивный | Отображение картинки при выделенной строке | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации»;  3. Перейти на вкладку «Ветеринарное свидетельство»; | Ожидаемый результат  При выделении строки отобразиться рисунок. |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат представлен на рисунке 5.7. |

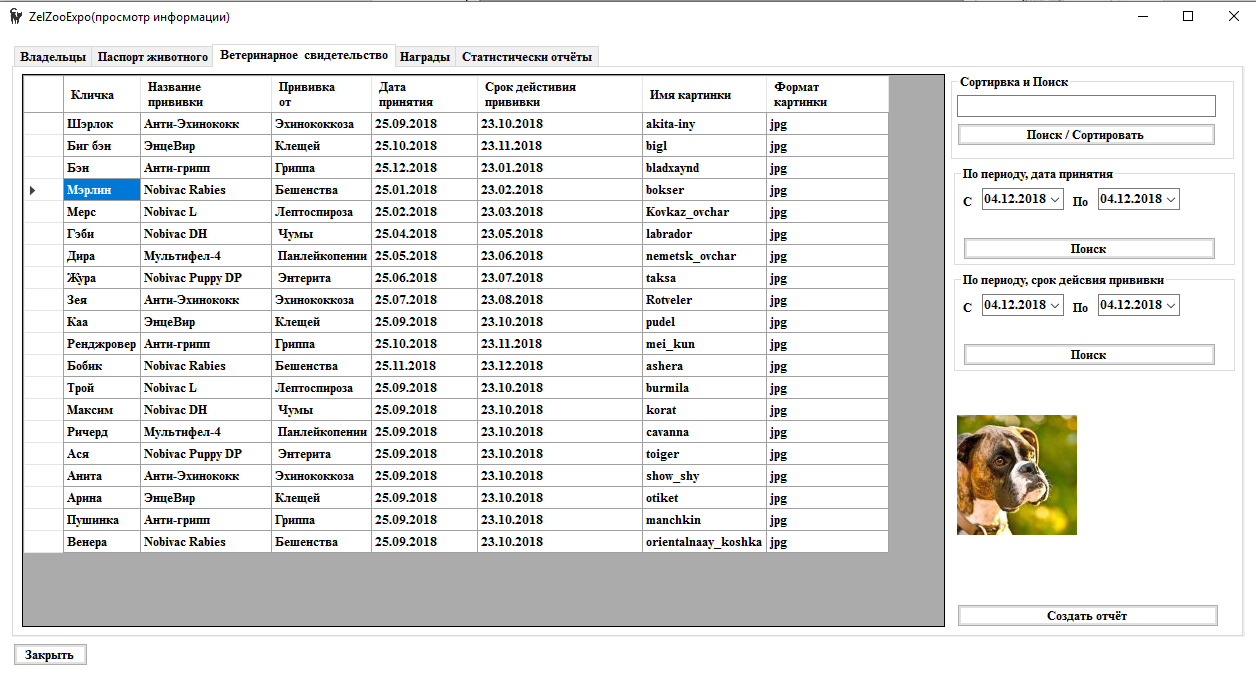


Рисунок 5.7

Таблица 5.8 – Test-case для контроля календаря прививок

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 008 | Позитивный | Контроль сроков прививки | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации». | Ожидаемый результат  При заходе, будет предупреждение что срок годности прививки истёк. |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат контроля прививок представлена на рисунке 5.8. |

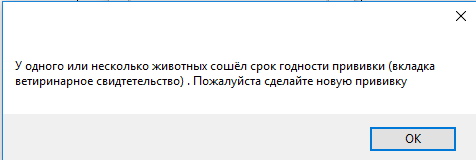


Рисунок 5.8

Таблица 5.9 – Test-case для создания отчётов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| 009 | Позитивный | Создание отчётов при нажатии на кнопку «Создать отчёт» | 1. Запустить приложение «ZelZooExpo.exe»;  2. Нажать на кнопку «Просмотр информации»;  3. Зайти на любую вкладку  4. Нажать кнопку «Создать отчёт». | Ожидаемый результат  При нажатии на кнопку создаться отчёт по выбранным данным |
| Фактический результат  Результат соответствует ожидаемому. Все условия соблюдены. Результат создания отчёта представлен на рисунке 5.9. |

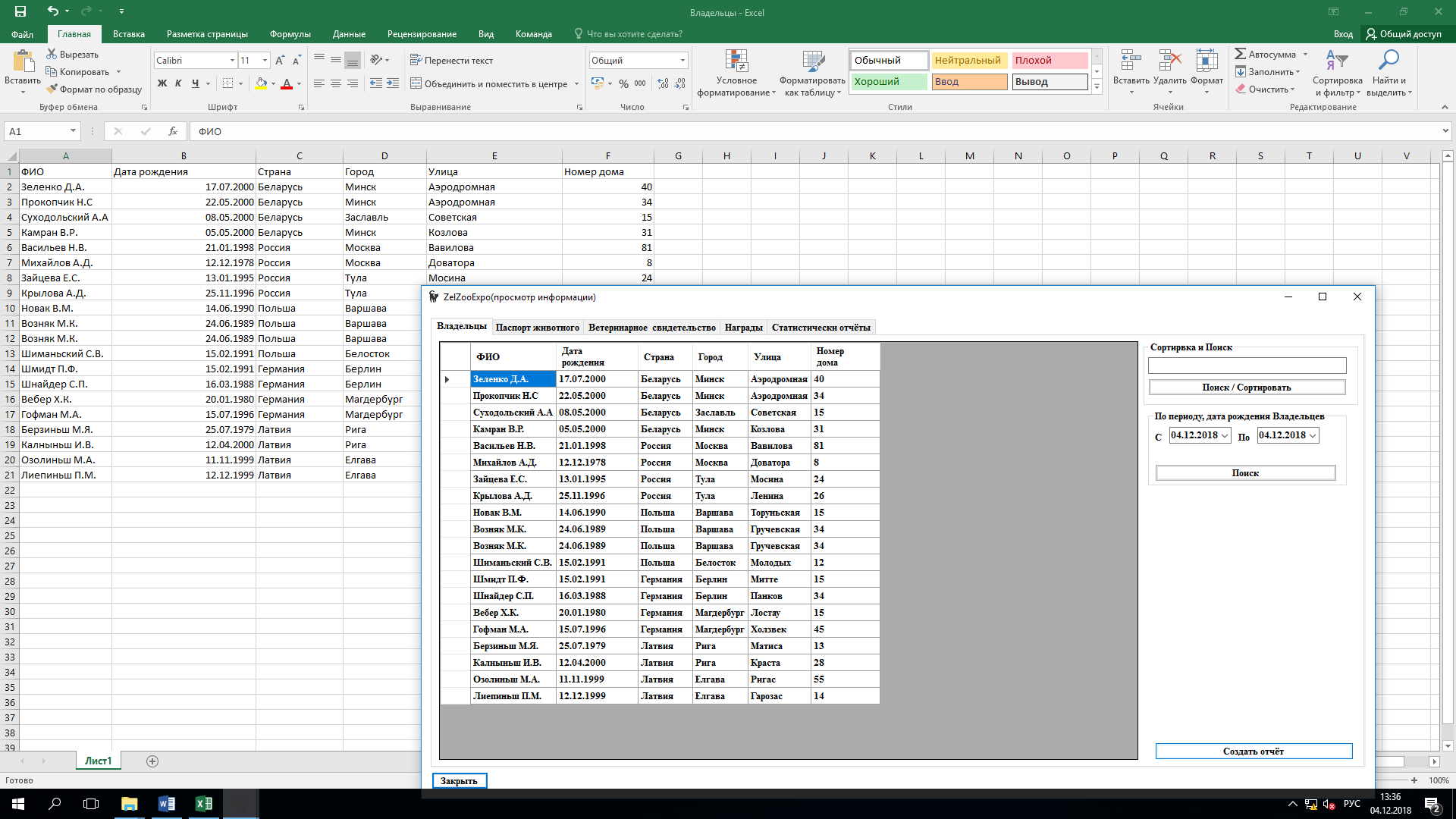


Рисунок 5.9

# 6 Применение

## 6.1 Назначение программы

Программа для автоматизации учёта информации о прививках и наградах животных «ZelZooExpo», предназначена для оперативного учёта животных, их наградах, прививках, владельцах. В этой программе возможно составлять отчёты по различным критериям, так же и статистические отчёты о вакцинации животных и их наградах.

## 6.2 Условия применения

Для применения данного программного средства необходимы следующие технически требования:

* процессор Intel Core 2 Duo или выше;
* минимальный объем оперативной памяти — 400 Мбайт;
* операционная система Windows 7 и выше;
* Framework v4.5 и выше
* устройство для чтения дисков.

Программа адаптирована под все компьютеры. Для переноса программы на другой компьютер достаточно будет перенести папку с проектом.

## 6.3 Справочная система

Справочная система программного средства представляет собой отдельный файл «ZelZooExpo\_Info.chm» с полным описанием основных функций программы в формате \*.chm. В справочной системе даны ответы на типичные вопросы, возникающие при работе с приложением, что, несомненно, должно помочь при освоении программного средства.

Справка имеет следующие разделы:

– «Введение»;

­– «Инструкция»;

– «Об авторе».

Структура справочной системы представлена на рисунке 6.1.

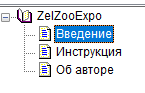


Рисунок 6.1

В разделе «Введение» рассказывается, что такое выставка животных. Содержимое раздела представлено на рисунке 6.2.

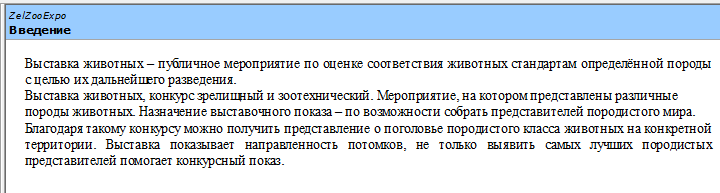


Рисунок 6.2

В разделе «Инструкция» приведена инструкция о работе с программой. Содержимое раздела представлено на рисунке 6.3.

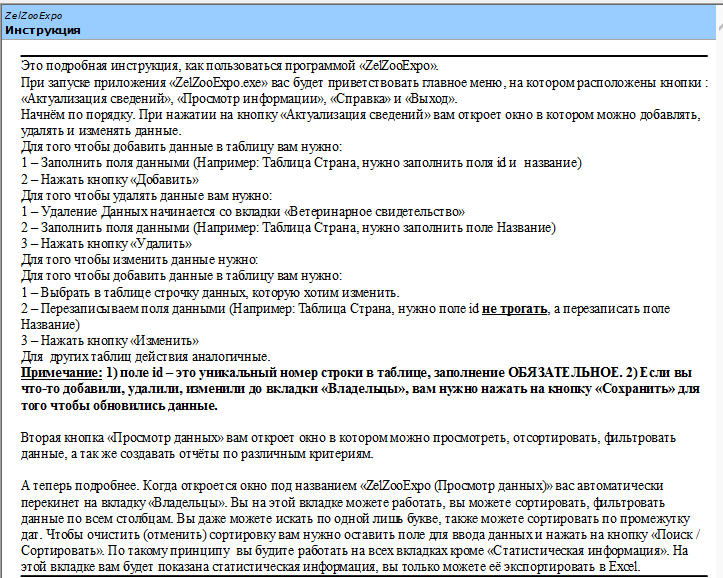


Рисунок 6.3

В разделе «Об авторе» отображена информация о разработчике программы. Содержание раздела представлено на рисунке 6.4.

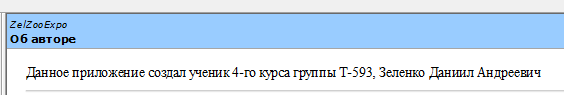


Рисунок 6.4

Заключение

В рамках курсового проектирования по дисциплине «Базы данных и системы управления базами данных» было разработано программное средство «ZelZooExpo.exe», автоматизирующее учёт информации о прививках и наградах животных.

Для достижения цели курсового проектирования были решены следующие задачи:

­– определена вычислительная система, необходимая для создания программного средства;

– разработаны логическая и физическая модели данных;

­– по модели выполнена проектирование задачи;

– разработано программное средство;

– описано созданное программное средство;

– выбрана методика испытания;

– описан процесс функционального тестирования;

– приведены примеры в области применения.

Разработка имеет интуитивно графический интерфейс, позволяющий с минимальным знанием компьютера использовать данное программное средство.

В процессе разработке данного программного средства были применены и закреплены знания по уже изученному материалу, были отработаны навыки владения методами надёжного программирования и эффективности разработки программного обеспечения в Visual studio 2017 с использованием языка программирования Visual Basic, разработана база данных средствами системы управления базами данных

Программа реализована в полном объёме и в соответствии с техническим заданием, полностью отложено и протестирована. Поставленные задачи выполнены.

Программа готова к практическому использованию в компании «ZelZooExpo».

# Список информационных источников

1. Орлов, С.А. Технологии разработки программного обеспечения: Учебник для вузов. 4-е изд. / С. А. Орлов, Б. Я. Цилькер – СПБ. : Питер, 2012.- 608 с.
2. Тепляков, С. Паттерны проектирования на платформе .NET. / С. Тепляков – СПБ. : Питер, 2015.-320 с.
3. Общие требования к текстовым документам: ГОСТ 2.105-95. – Введ. 01.01.1996 – Минск: Межгос. совет по стандартизации, метрологии и сертификации, 1995. – 84 с.
4. Программа и методика испытаний. Требования к содержанию, оформлению и контролю качества: ГОСТ 19.301. – 2000. – Введ. 01.01.2001. – Минск: Межгос. совет по стандартизации, метрологии и сертификации, 2000. – 14 с.
5. Текст программы. Требования к содержанию, оформлению и контролю качества: ГОСТ 19.401.-2000. – Введ. 01.09.2001. – Минск: Межгос. совет по стандартизации, метрологии и сертификации, 2000. – 16 с.
6. Windows 10[электронный ресурс]. Windows 10 характеристики – Режим доступа: <https://ru.wikipedia.org/wiki/Windows_10> - Дата доступа: 11.12.2018.
7. Microsoft Office [электронный ресурс]. Microsoft Excel офисный пакет– Режим доступа: <https://ru.wikipedia.org/wiki/Microsoft_Excel> – Дата доступа: 11.12.2018.
8. Microsoft SQL Server 2017 [электронный ресурс]. Система управления базами данных – Режим доступа: <https://ru.wikipedia.org/wiki/SQL_Server> – Дата доступа: 11.12.2018.
9. Microsoft Visual Studio 2017 [электронный ресурс]. Среда разработки – Режим доступа: <https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio> – Дата доступа: 11.12.2018.
10. Visual Basic [электронный ресурс]. Язык программирования – Режим доступа: https://ru.wikipedia.org/wiki/ Visual\_Basic – Дата доступа: 11.12.2018.
11. Help and manual [электронный ресурс]. Среда разработки справочной системы – Режим доступа: <https://en.wikipedia.org/wiki/Help_%26_Manual> – Дата доступа: 11.12.2018.
12. UMLet [электронный ресурс]. Среда разработки диаграмм – Режим доступа: <https://en.wikipedia.org/wiki/UMLet> – Дата доступа: 11.12.2018.
13. Smart install maker [электронный ресурс]. Среда разработки инсталлятора – Режим доступа: <https://ru.wikipedia.org/wiki/Smart_Install_Maker> – Дата доступа: 11.12.2018.
14. Sql server management studio [электронный ресурс]. Среда разработки базы данных – Режим доступа: <https://ru.wikipedia.org/wiki/SQL_Server_Management_Studio> – Дата доступа

Приложение А

(обязательное)

Тест программных модулей

create database animalBD

use animalBD

//создание таблиц

create table Страна

(ID\_Страны int not null,Название nvarchar(20) not null,constraint PK\_ID\_Страны primary key (ID\_Страны))

create table Награды

(ID\_Награды int not null, Наименование nvarchar(100) not null,Количество\_животных\_получивших\_награду int null, constraint PK\_ID\_Награды primary key (ID\_Награды) )

create table ОтчегоПрививка

(ID\_отЧего int not null,Прививка\_от nvarchar(20) not null,constraint PK\_ID\_отЧего primary key (ID\_отЧего))

-- drop table ОтчегоПрививка

create table Прививка(ID\_прив int not null,Название\_прививки nvarchar(20) not null,Страна\_производитель nvarchar(20) not null,Количество\_привитых int null,constraint PK\_ID\_прив primary key (ID\_прив))

-- drop table Прививка

create table Место\_размещения\_электронного\_чипа

(ID\_Место int not null,Где nvarchar(20) not null,constraint PK\_ID\_Место primary key (ID\_Место))

-- drop table Место\_размещения\_электронного\_чипа

create table Породы

(ID\_Породы int not null,Название nvarchar(20) not null,constraint PK\_ID\_Породы primary key (ID\_Породы))

--drop table Пароды

create table Владельцы

(ID\_Владельца int not null,ФИО nvarchar(30) not null,Дата\_рождения date not null,Страна int not null,Город nvarchar(20) not null,Улица nvarchar(20) not null,Номер\_дома int not null,constraint PK\_ID\_Владельца primary key (ID\_Владельца),constraint FK\_Страна foreign key (Страна) references Страна(ID\_Страны))

--drop table Владельцы

create table Паспорт\_животного

(ID\_Животного int not null,Владелец int not null,Номер\_родословной int not null,Кличка nvarchar(20) not null,Порода int not null,Дата\_рождения date not null,Группа nvarchar(20) not null,Окрас nvarchar(20) not null,Пол nvarchar(1) not null,Имя\_мамы nvarchar(20) not null,Имя\_папы nvarchar(20) not null,Номер\_чипа int not null,Дата\_имплантации date not null,Место\_размещения int not null,constraint PK\_ID\_Животного primary key (ID\_Животного),constraint FK\_Владелец foreign key (Владелец) references Владельцы(ID\_Владельца),constraint FK\_Порода foreign key (Порода) references Породы(ID\_Породы),constraint FK\_Место\_размещения foreign key (Место\_размещения)references Место\_размещения\_электронного\_чипа(ID\_Место))

-- drop table Паспорт\_животного

create table Получение\_награды

(ID\_Получ int not null,Кличка\_животного int not null,Наименование\_награды int not null,Дата\_получения date not null,Прошлая\_дата\_получения date not null,constraint PK\_ID\_Получ primary key (ID\_Получ),constraint FK\_Кличка\_животного foreign key (Кличка\_животного) references Паспорт\_животного(ID\_Животного),constraint FK\_Наименование\_награды foreign key (Наименование\_награды) references Награды(ID\_Награды))

--drop table Получение\_награды

create table Ветеринарное\_свидетельство

(ID\_Вет int not null,Имя\_животного int not null,Наименование\_прививки int not null,Привика\_от int not null,Дата\_принятия date not null,Срок\_действия\_прививки date not null,Имя\_картинки nvarchar(20) not null,

Формат\_картинки nvarchar(4) not null,constraint PK\_ID\_Вет primary key (ID\_Вет),

constraint FK\_Имя\_животного foreign key (Имя\_животного) references Паспорт\_животного(ID\_Животного),

constraint FK\_Наименование\_прививки foreign key (Наименование\_прививки) references Прививка(ID\_прив),

constraint FK\_Привика\_от foreign key (Привика\_от) references ОтчегоПрививка(ID\_отЧего))

--drop table Ветеринарное\_свидетельство

//создание триггеров

Create trigger trigger\_add\_info\_NAGRAD on Получение\_награды for insert as update Награды

set Количество\_животных\_получивших\_награду = s.Количество\_животных\_получивших\_награду +1

from Награды s join inserted i on s.ID\_Награды = i.Наименование\_награды

Create trigger trigger\_del\_info\_NAGRAD on Получение\_награды for delete as update Награды

set Количество\_животных\_получивших\_награду = s.Количество\_животных\_получивших\_награду -1

from Награды s join deleted d on s.ID\_Награды = d.Наименование\_награды

Create trigger trigger\_add\_info\_PRIV on Ветеринарное\_свидетельство for insert as update Прививка

set Количество\_привитых = s.Количество\_привитых +1 from Прививка s join inserted i

on s.ID\_прив = i.Наименование\_прививки

Create trigger trigger\_del\_info\_PRIV on Ветеринарное\_свидетельство for delete as update Прививка

set Количество\_привитых = s.Количество\_привитых -1 from Прививка s join deleted d on s.ID\_прив = d.Наименование\_прививки

/\*заполнение таблицы Страна\*/

insert into Страна(ID\_Страны, Название) values (1,N'Беларусь')

insert into Страна(ID\_Страны, Название) values (2,N'Россия')

insert into Страна(ID\_Страны, Название) values (3,N'Польша')

insert into Страна(ID\_Страны, Название) values (4,N'Германия')

insert into Страна(ID\_Страны, Название) values (5,N'Латвия')

/\*заполнение таблицы Награды\*/

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (1,N'Награда за хорошее выполнение команд',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (2,N'Награда за лучший номер',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (3,N'Награда за быстрое преодоление препятствий',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (4,N'Лучший щенок породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (5,N'Лучший кот породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (6,N'Лучший юниор',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (7,N'Лучший юниор породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (8,N'Победитель класса',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (9,N'Лучший ветеран',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (10,N'Лучший ветеран породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (11,N'Лучшая сука породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (12,N'Лучший кобель породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (13,N'Лучший представитель породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (14,N'Лучший в группе',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (15,N'Лучшая собака выставки',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (16,N'Лучшая кошка выставки',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (17,N'Награда за красоту породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (18,N'Награда за милату породы',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (19,N'Награда за милату котят',0)

insert into Награды(ID\_Награды,Наименование,Количество\_животных\_получивших\_награду)

values (20,N'Награда за милату щенят',0)

/\*заполенеие таблицы отчего прививка\*/

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (1,N'Эхинококкоза')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (2,N'Клещей')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (3,N'Гриппа')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (4,N'Бешенства')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (5,N'Лептоспироза')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (6,N' Чумы')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (7,N' Панлейкопении')

insert into ОтчегоПрививка(ID\_отЧего,Прививка\_от) values (8,N' Энтерита')

/\*Заполнение таблицы Прививка\*/

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (1,N'Анти-Эхинококк',N'Россия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (2,N'ЭнцеВир',N'Россия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (3,N'Анти-грипп',N'Россия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (4,N'Nobivac Rabies',N'Голландия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (5,N'Nobivac L',N'Голландия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (6,N'Nobivac DH',N'Голландия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (7,N'Мультифел-4',N'Россия',0)

insert into Прививка(ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых)

values (8,N'Nobivac Puppy DP',N'Голландия',0)

/\*Заполнение таблицы Место\_размещения\_электронного\_чипа\*/

insert into Место\_размещения\_электронного\_чипа(ID\_Место,Где) values (1,N'Шея')

insert into Место\_размещения\_электронного\_чипа(ID\_Место,Где) values (2,N'Крестцовая область')

insert into Место\_размещения\_электронного\_чипа(ID\_Место,Где) values (3,N'Между лопатками')

/\*Заполнение таблицы пароды\*/

insert into Породы(ID\_Породы,Название) Values (1,N'Акита-ину')

insert into Породы(ID\_Породы,Название) Values (2,N'Бигль')

insert into Породы(ID\_Породы,Название) Values (3,N'Бладхаунд')

insert into Породы(ID\_Породы,Название) Values (4,N'Боксёр')

insert into Породы(ID\_Породы,Название) Values (5,N'Кавказкая овчарка')

insert into Породы(ID\_Породы,Название) Values (6,N'Лабрадор')

insert into Породы(ID\_Породы,Название) Values (7,N'Немецкая\_овчарка')

insert into Породы(ID\_Породы,Название) Values (8,N'Такса')

insert into Породы(ID\_Породы,Название) Values (9,N'Ротвейлер')

insert into Породы(ID\_Породы,Название) Values (10,N'Пудель')

insert into Породы(ID\_Породы,Название) Values (11,N'Мэйн-кун')

insert into Породы(ID\_Породы,Название) Values (12,N'Ашера')

insert into Породы(ID\_Породы,Название) Values (13,N'Бурмила')

insert into Породы(ID\_Породы,Название) Values (14,N'Корат')

insert into Породы(ID\_Породы,Название) Values (15,N'Саванна')

insert into Породы(ID\_Породы,Название) Values (16,N'Тойгер')

insert into Породы(ID\_Породы,Название) Values (17,N'Сноу шу')

insert into Породы(ID\_Породы,Название) Values (18,N'Оцикет')

insert into Породы(ID\_Породы,Название) Values (19,N'Манчкин')

insert into Породы(ID\_Породы,Название) Values (20,N'Ориентальная кошка')

/\*Заполнение таблицы Владельцы\*/

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома) /\*ZAPOLNENO\*/

values (1,N'Зеленко Д.А.','2000-07-17',1,N'Минск',N'Аэродромная',40)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (2,N'Прокопчик Н.С','2000-05-22',1,N'Минск',N'Аэродромная',34)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (3,N'Суходольский А.А','2000-05-08',1,N'Заславль',N'Советская',15)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (4,N'Камран В.Р.','2000-05-05',1,N'Минск',N'Козлова',31)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (5,N'Васильев Н.В.','1998-01-21',2,N'Москва',N'Вавилова',81)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (6,N'Михайлов А.Д.','1978-12-12',2,N'Москва',N'Доватора',8)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (7,N'Зайцева Е.С.','1995-01-13',2,N'Тула',N'Мосина',24)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (8,N'Крылова А.Д.','1996-11-25',2,N'Тула',N'Ленина',26)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (9,N'Новак В.М.','1990-06-14',3,N'Варшава',N'Торуньская',15)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (10,N'Возняк М.К.','1989-06-24',3,N'Варшава',N'Гручевская',34)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (11,N'Возняк М.К.','1989-06-24',3,N'Варшава',N'Гручевская',34)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (12,N'Шиманьский С.В.','1991-02-15',3,N'Белосток',N'Молодых',12)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (13,N'Шмидт П.Ф.','1991-02-15',4,N'Берлин',N'Митте',15)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (14,N'Шнайдер С.П.','1988-03-16',4,N'Берлин',N'Панков',34)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (15,N'Вебер Х.К.','1980-01-20',4,N'Магдербург',N'Лостау',15)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (16,N'Гофман М.А.','1996-07-15',4,N'Магдербург',N'Холзвек',45)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (17,N'Берзиньш М.Я.','1979-07-25',5,N'Рига',N'Матиса',13)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (18,N'Калныньш И.В.','2000-04-12',5,N'Рига',N'Краста',28)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (19,N'Озолиньш М.А.','1999-11-11',5,N'Елгава',N'Ригас',55)

insert into Владельцы(ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома)

values (20,N'Лиепиньш П.М.','1999-12-12',5,N'Елгава',N'Гарозас',14)

/\*Заполнение таблицы Паспорт\_животного\*/

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (1,20151,1,N'Собака',N'Шэрлок',1,'2012-05-05',N'Белый',N'М',N'Агата',N'Амур','2012-08-15',3245,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (2,21547,2,N'Собака',N'Биг бэн',2,'2012-06-04',N'Чёрный',N'М',N'Бада',N'Рио','2012-10-10',3455,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (3,26548,3,N'Собака',N'Бэн',3,'2013-04-07',N'Серый',N'М',N'Вайда',N'Нарзан','2013-10-10',4785,2)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (4,24875,4,N'Собака',N'Мэрлин',4,'2013-10-17',N'Тёмно-серый',N'М',N'Волга',N'Пегас','2014-01-10',5548,2)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (5,27834,5,N'Собака',N'Мерс',5,'2012-10-10',N'Светло-серый',N'М',N'Гизель',N'Принц','2013-01-10',8754,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (6,24579,6,N'Собака',N'Гэби',6,'2012-12-10',N'Светло-серый',N'Ж',N'Дымка',N'Клайд','2013-01-10',9877,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (7,45784,7,N'Собака',N'Дира',7,'2013-01-10',N'Белый',N'Ж',N'Жуча',N'Лео','2013-09-01',7777,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (8,31548,8,N'Собака',N'Жура',8,'2015-05-10',N'Белый',N'Ж',N'Жуча',N'Лукас','2015-09-11',6666,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (9,78542,9,N'Собака',N'Зея',9,'2012-05-10',N'Чёрный',N'Ж',N'Жуч',N'Ларион','2015-09-11',5555,2)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (10,59789,10,N'Собака',N'Каа',10,'2014-05-10',N'Чёрный',N'Ж',N'Жуч',N'Ларион','2015-10-05',4444,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (11,65775,11,N'Кошка',N'Ренджровер',11,'2016-08-11',N'Чёрный',N'М',N'Супер',N'Спорт','2016-10-05',1111,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (12,67158,12,N'Кошка',N'Бобик',12,'2015-04-12',N'Рыжий',N'М',N'Мила',N'Тритон','2015-10-25',1441,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (13,46482,13,N'Кошка',N'Трой',13,'2015-04-13',N'Рыжий',N'М',N'Ли-ли',N'Марс','2015-02-24',1548,2)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (14,87546,14,N'Кошка',N'Максим',14,'2016-04-13',N'Серый',N'М',N'Линда',N'Ричерд','2016-09-29',5748,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (15,53467,15,N'Кошка',N'Ричерд',15,'2012-05-21',N'Белый',N'М',N'Луя',N'Луиджи','2012-12-20',3415,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (16,46317,16,N'Кошка',N'Ася',16,'2013-05-21',N'Рыжий',N'Ж',N'Дайра',N'Баста','2013-12-20',4785,2)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (17,31548,17,N'Кошка',N'Анита',17,'2012-06-11',N'Рыжий',N'Ж',N'Багира',N'Баки','2012-12-20',4615,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (18,31548,18,N'Кошка',N'Арина',18,'2012-07-19',N'Белый',N'Ж',N'Ася',N'Джэк','2012-12-12',4567,1)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (19,89548,19,N'Кошка',N'Пушинка',19,'2012-07-21',N'Чёрный',N'Ж',N'Атика',N'Вафля','2013-01-10',7646,3)

insert into Паспорт\_животного(ID\_Животного,Номер\_родословной,Владелец,Группа,Кличка,Порода,Дата\_рождения,Окрас,Пол,Имя\_мамы,Имя\_папы,Дата\_имплантации,Номер\_чипа,Место\_размещения)

values (20,78453,20,N'Кошка',N'Венера',19,'2012-07-21',N'Серый',N'Ж',N'Веста',N'Боря','2013-01-10',4567,2)

/\*Заполнение таблицы Получение\_нарграды\*/

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(1,1,1,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(2,2,2,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(3,3,3,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(4,4,4,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(5,5,5,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(6,6,6,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(7,7,7,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(8,8,8,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(9,9,9,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(10,10,10,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(11,11,11,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(12,12,12,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(13,13,13,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(14,14,14,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(15,15,15,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(16,16,16,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(17,17,17,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(18,18,18,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(19,19,19,'2018-09-25','2017-09-25')

insert into Получение\_награды(ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) values(20,20,20,'2018-09-25','2017-09-25')

/\*Заполнение таблицы Ветеринарное свидетельство\*/

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(1,1,1,1,'2018-09-25','2018-10-23','akita-iny','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(2,2,2,2,'2018-10-25','2018-11-23','bigl','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(3,3,3,3,'2018-12-25','2018-01-23','bladxaynd','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(4,4,4,4,'2018-01-25','2018-02-23','bokser','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(5,5,5,5,'2018-02-25','2018-03-23','Kovkaz\_ovchar','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки)values(6,6,6,6,'2018-04-25','2018-05-23','labrador','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки)values(7,7,7,7,'2018-05-25','2018-06-23','nemetsk\_ovchar','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки)

values(8,8,8,8,'2018-06-25','2018-07-23','taksa','jpg')insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(9,9,1,1,'2018-07-25','2018-08-23','Rotveler','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(10,10,2,2,'2018-09-25','2018-10-23','pudel','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(11,11,3,3,'2018-10-25','2018-11-23','mei\_kun','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(12,12,4,4,'2018-11-25','2018-12-23','ashera','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(13,13,5,5,'2018-09-25','2018-10-23','burmila','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки)

values(14,14,6,6,'2018-09-25','2018-10-23','korat','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(15,15,7,7,'2018-09-25','2018-10-23','cavanna','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(16,16,8,8,'2018-09-25','2018-10-23','toiger','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(17,17,1,1,'2018-09-25','2018-10-23','show\_shy','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(18,18,2,2,'2018-09-25','2018-10-23','otiket','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(19,19,3,3,'2018-09-25','2018-10-23','manchkin','jpg')

insert into Ветеринарное\_свидетельство(ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) values(20,20,4,4,'2018-09-25','2018-10-23','orientalnaay\_koshka','jpg')

// начальная форма

Public Class StartForm

Private Sub update\_INFO\_Click(sender As Object, e As EventArgs) Handles update\_INFO.Click

aktyalizat.Show() End Sub

Private Sub Read\_INFO\_Click(sender As Object, e As EventArgs) Handles Read\_INFO.Click

prosmotr.Show() End Sub

Private Sub Spravka\_Click(sender As Object, e As EventArgs) Handles Spravka.Click

Process.Start("ZelZooExpo\_Info.chm") End Sub

Private Sub Exit\_app\_Click(sender As Object, e As EventArgs) Handles Exit\_app.Click

Application.Exit()

End Sub

Private Sub StartForm\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

MessageBox.Show("Добро пожаловать в программу ZelZooExpo. " & vbCrLf & "Пожалуйста перед использованием программы прочтите СПРАВКУ. " & vbCrLf & "Спарвку можно вызвать клавишей F1 или же при нажатии на кнопку СПРАВКА.")

KeyPreview = True End Sub

Private Sub StartForm\_KeyDown(sender As Object, e As KeyEventArgs) Handles MyBase.KeyDown

If e.KeyCode = Keys.F1 Then Help.ShowHelp(Me, "ZelZooExpo\_Info.chm") End Sub End Class

// актуализация сведений

Imports System.Configuration Imports System.Data Imports System.Data.SqlClient

Public Class aktyalizat

Private Sub Vixod\_Click(sender As Object, e As EventArgs) Handles Vixod.Click

Close() End Sub

Private Sub aktyalizat\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

'TODO: данная строка кода позволяет загрузить данные в таблицу "AnimalBDDataSet.Паспорт\_животного". При необходимости она может быть перемещена или удалена.

Me.Паспорт\_животногоTableAdapter.Fill(Me.AnimalBDDataSet.Паспорт\_животного)

Me.ВладельцыTableAdapter.Fill(Me.AnimalBDDataSet.Владельцы)

Me.ПородыTableAdapter.Fill(Me.AnimalBDDataSet.Породы) Me.Место\_размещения\_электронного\_чипаTableAdapter.Fill(Me.AnimalBDDataSet.Место\_размещения\_электронного\_чипа)

Me.ПрививкаTableAdapter.Fill(Me.AnimalBDDataSet.Прививка)

Me.ОтчегоПрививкаTableAdapter.Fill(Me.AnimalBDDataSet.ОтчегоПрививка)

Me.НаградыTableAdapter.Fill(Me.AnimalBDDataSet.Награды)

Me.СтранаTableAdapter.Fill(Me.AnimalBDDataSet.Страна)

DGV\_Страна\_актул.DataSource = Getdatalist()

DGV\_Награды\_актул.DataSource = Getdatalist1()

DGV\_отчего\_актул.DataSource = Getdatalist2()

DGV\_прививки\_актул.DataSource = Getdatalist3()

DGV\_место\_размещ\_актул.DataSource = Getdatalist4()

DGV\_породы\_актул.DataSource = Getdatalist5()

DGV\_Владельцы\_актул.DataSource = Getdatalist6()

DGV\_паспорт\_животного\_актул.DataSource = Getdatalist7()

DGV\_получ\_награды\_актул.DataSource = Getdatalist8()

DGV\_ветеринарное\_свидетельство.DataSource = Getdatalist9()

KeyPreview = True End Sub

Private Sub save\_Click(sender As Object, e As EventArgs) Handles save.Click

MessageBox.Show("Внимание приложение будет презапущено") Application.Restart()End Sub

Private Sub Vladelti\_Click(sender As Object, e As EventArgs) Handles Vladelti.Click

MessageBox.Show("Внимание нажмите кнопку СОХРАНИТЬ")End Sub

Private Sub Добавить\_страну\_Click(sender As Object, e As EventArgs) Handles Добавить\_страну.Click

If id\_страны\_доб.Text <> "" And название\_страны\_доб.Text <> "" ThenDim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)Dim query = "INSERT INTO Страна (ID\_Страны,Название) VALUES(@id\_str, @NazvStr)"Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_str", id\_страны\_доб.Text)

cmd.Parameters.AddWithValue("@NazvStr", название\_страны\_доб.Text)conn.Open()cmd.ExecuteNonQuery()conn.Close()End UsingEnd Using

id\_страны\_доб.Clear() : название\_страны\_доб.Clear() : DGV\_Страна\_актул.DataSource = Getdatalist()

Else MessageBox.Show("Заполните поле(я) данными")End IfEnd Sub

Private Sub Удалить\_страну\_Click(sender As Object, e As EventArgs) Handles Удалить\_страну.Click

If название\_страны\_удл.Text <> "" Then Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString) Dim query = "delete from Страна where Название = @NazvStr\_d"

Using cmd As New SqlCommand(query, conn) cmd.Parameters.AddWithValue("@NazvStr\_d", название\_страны\_удл.Text) conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

название\_страны\_удл.Clear() : DGV\_Страна\_актул.DataSource = Getdatalist()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Изменить\_страну\_Click(sender As Object, e As EventArgs) Handles Изменить\_страну.Click

If id\_страны\_обнв.Text <> "" And название\_страны\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Страна set Название=@NazvStr\_up where ID\_Страны= @id\_str\_up"

Using cmd As New SqlCommand(query, conn) cmd.Parameters.AddWithValue("@id\_str\_up", id\_страны\_обнв.Text) cmd.Parameters.AddWithValue("@NazvStr\_up", название\_страны\_обнв.Text) conn.Open()

cmd.ExecuteNonQuery() conn.Close() End Using End Using

id\_страны\_обнв.Clear() : название\_страны\_обнв.Clear() : DGV\_Страна\_актул.DataSource = Getdatalist()

Else MessageBox.Show("Заполните поле(я) данными")End IfEnd Sub

Private Function Getdatalist() As DataTable

Dim dt As New DataTable Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from Страна", conn)

conn.Open() : Dim reader As SqlDataReader = cmd.ExecuteReader() : dt.Load(reader)

End Using End Using Return dt End Function

Private Sub Добавить\_награду\_Click(sender As Object, e As EventArgs) Handles Добавить\_награду.Click

If id\_награды\_доб.Text <> "" And наименование\_награды\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Награды (ID\_Награды,Наименование,Количество\_животных\_получивших\_награду) VALUES(@id\_nag, @NazvNag, @kolVoNag)"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_nag", id\_награды\_доб.Text)

cmd.Parameters.AddWithValue("@NazvNag", наименование\_награды\_доб.Text)

cmd.Parameters.AddWithValue("@kolVoNag", колживнаг\_доб.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

колживнаг\_доб.Clear() : id\_награды\_доб.Clear() : наименование\_награды\_доб.Clear() : DGV\_Награды\_актул.DataSource = Getdatalist1()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Удалить\_награду\_Click(sender As Object, e As EventArgs) Handles Удалить\_награду.Click

If наименование\_награды\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Награды where Наименование = @NazvNag\_d"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@NazvNag\_d", наименование\_награды\_дел.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

наименование\_награды\_дел.Clear() : DGV\_Награды\_актул.DataSource = Getdatalist1()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Изменить\_награду\_Click(sender As Object, e As EventArgs) Handles Изменить\_награду.Click

If id\_награды\_обнв.Text <> "" And наименование\_награды\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Награды set Наименование=@NazvNag\_up, Количество\_животных\_получивших\_награду=@kolvonag2 where ID\_Награды= @id\_nag\_up"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_nag\_up", id\_награды\_обнв.Text)

cmd.Parameters.AddWithValue("@NazvNag\_up", наименование\_награды\_обнв.Text)

cmd.Parameters.AddWithValue("@kolvonag2", колживнаг\_обнов.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

колживнаг\_обнов.Clear() : id\_награды\_обнв.Clear() : наименование\_награды\_обнв.Clear() : DGV\_Награды\_актул.DataSource = Getdatalist1()

Else MessageBox.Show("Заполните поле(я) данными")End If

End Sub

Private Function Getdatalist1() As DataTable Dim dt As New DataTable Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from Награды", conn)

conn.Open() : Dim reader As SqlDataReader = cmd.ExecuteReader() : dt.Load(reader)

End Using End Using Return dt

End Function

Private Sub Добавить\_отчего\_прививка\_Click(sender As Object, e As EventArgs) Handles Добавить\_отчего\_прививка.Click

If id\_отчего\_прив\_доб.Text <> "" And прививка\_от\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO ОтчегоПрививка (ID\_отЧего,Прививка\_от) VALUES(@id\_otchego, @prirviv\_ot)"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_otchego", id\_отчего\_прив\_доб.Text)

cmd.Parameters.AddWithValue("@prirviv\_ot", прививка\_от\_доб.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

id\_отчего\_прив\_доб.Clear() : прививка\_от\_доб.Clear() : DGV\_отчего\_актул.DataSource = Getdatalist2()

Else MessageBox.Show("Заполните поле(я) данными") End If

End Sub

Private Sub Удалить\_отчего\_прививка\_Click(sender As Object, e As EventArgs) Handles Удалить\_отчего\_прививка.Click If прививка\_от\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from ОтчегоПрививка where Прививка\_от = @prirviv\_ot\_d"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@prirviv\_ot\_d", прививка\_от\_дел.Text) conn.Open()

cmd.ExecuteNonQuery() conn.Close() End Using

End Using прививка\_от\_дел.Clear() : DGV\_отчего\_актул.DataSource = Getdatalist2()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Изменить\_отчего\_прививка\_Click(sender As Object, e As EventArgs) Handles Изменить\_отчего\_прививка.Click

If id\_отчего\_прив\_обнв.Text <> "" And прививка\_от\_обнв.Text <> "" Then Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update ОтчегоПрививка set Прививка\_от=@prirviv\_ot\_up where ID\_отЧего= @id\_otchego\_up"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_otchego\_up", id\_отчего\_прив\_обнв.Text)

cmd.Parameters.AddWithValue("@prirviv\_ot\_up", прививка\_от\_обнв.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

id\_отчего\_прив\_обнв.Clear() : прививка\_от\_обнв.Clear() : DGV\_отчего\_актул.DataSource = Getdatalist2()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub Private Function Getdatalist2() As DataTable Dim dt As New DataTable Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from ОтчегоПрививка", conn) conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader() dt.Load(reader) End Using End Using Return dt

End Function

Private Sub Добавить\_прививку\_Click(sender As Object, e As EventArgs) Handles Добавить\_прививку.Click

If id\_прививки\_доб.Text <> "" And наименование\_прививки\_доб.Text <> "" And страна\_производитель\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Прививка (ID\_прив,Название\_прививки,Страна\_производитель,Количество\_привитых) VALUES(@ID\_priv\_ad, @nazvpriv\_ad,@strana\_proizv\_ad,@kolovopriv)"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@ID\_priv\_ad", id\_прививки\_доб.Text)

cmd.Parameters.AddWithValue("@nazvpriv\_ad", наименование\_прививки\_доб.Text)

cmd.Parameters.AddWithValue("@strana\_proizv\_ad", страна\_производитель\_доб.Text)

cmd.Parameters.AddWithValue("@kolovopriv", колвоприв\_доб.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

колвоприв\_доб.Clear() : id\_прививки\_доб.Clear() : наименование\_прививки\_доб.Clear() : страна\_производитель\_доб.Clear() : DGV\_прививки\_актул.DataSource = Getdatalist3()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Удалить\_прививку\_Click(sender As Object, e As EventArgs) Handles Удалить\_прививку.Click

If наименование\_прививки\_дел.Text <> "" Then Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString) Dim query = "delete from Прививка where Название\_прививки = @nazvpriv\_d" Using cmd As New SqlCommand(query, conn) cmd.Parameters.AddWithValue("@nazvpriv\_d", наименование\_прививки\_дел.Text) conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

наименование\_прививки\_дел.Clear() : DGV\_прививки\_актул.DataSource = Getdatalist3() Else

MessageBox.Show("Заполните поле(я) данными")End If End Sub

Private Sub Изменить\_прививку\_Click(sender As Object, e As EventArgs) Handles Изменить\_прививку.Click

If id\_прививки\_обнв.Text <> "" And наименование\_прививки\_обнв.Text <> "" And страна\_производитель\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Прививка set Название\_прививки= @nazvpriv\_up , Страна\_производитель= @strana\_proizv\_up, Количество\_привитых=@kolvopriv where ID\_прив= @ID\_priv\_up"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@ID\_priv\_up", id\_прививки\_обнв.Text)

cmd.Parameters.AddWithValue("@nazvpriv\_up", наименование\_прививки\_обнв.Text)

cmd.Parameters.AddWithValue("@strana\_proizv\_up", страна\_производитель\_обнв.Text)

cmd.Parameters.AddWithValue("@kolvopriv", колвоприв\_обнов.Text)conn.Open()cmd.ExecuteNonQuery()conn.Close()

End Using End Using

колвоприв\_обнов.Clear() : id\_прививки\_обнв.Clear() : наименование\_прививки\_обнв.Clear() : страна\_производитель\_обнв.Clear() : DGV\_прививки\_актул.DataSource = Getdatalist3()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Function Getdatalist3() As DataTable Dim dt As New DataTable Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from Прививка", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader) End Using End Using Return dtEnd Function

Private Sub Добавить\_место\_размещ\_чипа\_Click(sender As Object, e As EventArgs) Handles Добавить\_место\_размещ\_чипа.Click

If id\_места\_доб.Text <> "" And где\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Место\_размещения\_электронного\_чипа (ID\_Место,Где) VALUES(@id\_mesta, @gde)"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_mesta", id\_места\_доб.Text)

cmd.Parameters.AddWithValue("@gde", где\_доб.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

id\_места\_доб.Clear() : где\_доб.Clear() : DGV\_место\_размещ\_актул.DataSource = Getdatalist4()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Удалить\_место\_размещ\_чипа\_Click(sender As Object, e As EventArgs) Handles Удалить\_место\_размещ\_чипа.Click

If где\_дел.Text <> "" Then Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Место\_размещения\_электронного\_чипа where Где = @gde\_d"

Using cmd As New SqlCommand(query, conn) cmd.Parameters.AddWithValue("@gde\_d", где\_дел.Text)

conn.Open() cmd.ExecuteNonQuery() conn.Close() End Using End Using

где\_дел.Clear() : DGV\_место\_размещ\_актул.DataSource = Getdatalist4() Else

MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Sub Изменить\_место\_размещ\_чипа\_Click(sender As Object, e As EventArgs) Handles Изменить\_место\_размещ\_чипа.Click

If id\_места\_обнв.Text <> "" And где\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Место\_размещения\_электронного\_чипа set Где=@gde\_up where ID\_Место= @id\_mesta\_up"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_mesta\_up", id\_места\_обнв.Text)

cmd.Parameters.AddWithValue("@gde\_up", где\_обнв.Text) conn.Open() cmd.ExecuteNonQuery() conn.Close()

End Using End Using

id\_места\_обнв.Clear() : где\_обнв.Clear() : DGV\_место\_размещ\_актул.DataSource = Getdatalist4()

Else MessageBox.Show("Заполните поле(я) данными") End If End Sub

Private Function Getdatalist4() As DataTable

Dim dt As New DataTable Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from Место\_размещения\_электронного\_чипа", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader) End UsingEnd Using Return dt

End Function

Private Sub Добавить\_породу\_Click(sender As Object, e As EventArgs) Handles Добавить\_породу.Click

If id\_породы\_доб.Text <> "" And название\_породы\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Породы (ID\_Породы,Название) VALUES(@id\_poroda, @nazvporod)"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_poroda", id\_породы\_доб.Text)

cmd.Parameters.AddWithValue("@nazvporod", название\_породы\_доб.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

id\_породы\_доб.Clear() : название\_породы\_доб.Clear() : DGV\_породы\_актул.DataSource = Getdatalist5()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Удалить\_породу\_Click(sender As Object, e As EventArgs) Handles Удалить\_породу.Click

If название\_породы\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Породы where Название = @nazvporod\_d"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@nazvporod\_d", название\_породы\_дел.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

название\_породы\_дел.Clear() : DGV\_породы\_актул.DataSource = Getdatalist5()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Изменить\_породу\_Click(sender As Object, e As EventArgs) Handles Изменить\_породу.Click

If id\_породы\_обнв.Text <> "" And название\_породы\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Породы set Название=@nazvporod\_up where ID\_Породы= @id\_poroda\_up"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@id\_poroda\_up", id\_породы\_обнв.Text)

cmd.Parameters.AddWithValue("@nazvporod\_up", название\_породы\_обнв.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

id\_породы\_обнв.Clear() : название\_породы\_обнв.Clear() : DGV\_породы\_актул.DataSource = Getdatalist5()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Function Getdatalist5() As DataTable

Dim dt As New DataTable

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select \* from Породы", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader)

End Using

End Using

Return dt

End Function

Private Sub Добавть\_владельца\_Click(sender As Object, e As EventArgs) Handles Добавть\_владельца.Click

If id\_владельца\_доб.Text <> "" And ФИО\_доб.Text <> "" And город\_доб.Text <> "" And улица\_доб.Text <> "" And номер\_дома\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Владельцы (ID\_Владельца,ФИО,Дата\_рождения,Страна,Город,Улица,Номер\_дома) VALUES(@id\_vlad,@fio,@data,@strana,@gorod,@ylitsa,@dom)"

Using cmd As New SqlCommand(query, conn)

id\_страны\_влад\_доб.SelectedIndex += 1

cmd.Parameters.AddWithValue("@id\_vlad", id\_владельца\_доб.Text)

cmd.Parameters.AddWithValue("@fio", ФИО\_доб.Text)

cmd.Parameters.AddWithValue("@data", D1.Value)

cmd.Parameters.AddWithValue("@strana", id\_страны\_влад\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@gorod", город\_доб.Text)

cmd.Parameters.AddWithValue("@ylitsa", улица\_доб.Text)

cmd.Parameters.AddWithValue("@dom", номер\_дома\_доб.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

id\_владельца\_доб.Clear() : ФИО\_доб.Clear() : город\_доб.Clear() : улица\_доб.Clear() : номер\_дома\_доб.Clear() : DGV\_Владельцы\_актул.DataSource = Getdatalist6()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub GroupBox4\_add\_Enter(sender As Object, e As EventArgs) Handles GroupBox4\_add.Enter

End Sub

Private Sub Удалить\_владельца\_Click(sender As Object, e As EventArgs) Handles Удалить\_владельца.Click

If ФИО\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Владельцы where ФИО = @fio\_d"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@fio\_d", ФИО\_дел.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

ФИО\_дел.Clear() : DGV\_Владельцы\_актул.DataSource = Getdatalist6()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Изменить\_владельца\_Click(sender As Object, e As EventArgs) Handles Изменить\_владельца.Click

If id\_владельца\_обнв.Text <> "" And ФИО\_обнв.Text <> "" And город\_обнв.Text <> "" And улица\_обнв.Text <> "" And номер\_дома\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Владельцы set ФИО =@fio\_u ,Дата\_рождения=@data\_u,Страна=@strana\_u,Город=@gorod\_u,Улица=@ylitsa\_u,Номер\_дома=@dom\_u where ID\_Владельца =@id\_vlad\_u"

Using cmd As New SqlCommand(query, conn)

id\_страны\_влад\_обнв.SelectedIndex += 1

cmd.Parameters.AddWithValue("@id\_vlad\_u", id\_владельца\_обнв.Text)

cmd.Parameters.AddWithValue("@fio\_u", ФИО\_обнв.Text)

cmd.Parameters.AddWithValue("@data\_u", D2.Value)

cmd.Parameters.AddWithValue("@strana\_u", id\_страны\_влад\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@gorod\_u", город\_обнв.Text)

cmd.Parameters.AddWithValue("@ylitsa\_u", улица\_обнв.Text)

cmd.Parameters.AddWithValue("@dom\_u", номер\_дома\_обнв.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

id\_владельца\_обнв.Clear() : ФИО\_обнв.Clear() : город\_обнв.Clear() : улица\_обнв.Clear() : номер\_дома\_обнв.Clear() : DGV\_Владельцы\_актул.DataSource = Getdatalist6()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Function Getdatalist6() As DataTable

Dim dt As New DataTable

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select Владельцы.ID\_Владельца as 'Номер Владельца', Владельцы.ФИО, Владельцы.Дата\_рождения as 'Дата рождения',Страна.Название as 'Страна',Владельцы.Город,Владельцы.Улица,Владельцы.Номер\_дома as 'Номер дома' from Владельцы join Страна on Владельцы.Страна=Страна.ID\_Страны", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader)

End Using

End Using

Return dt

End Function

Private Sub DGV\_Владельцы\_актул\_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DGV\_Владельцы\_актул.CellContentClick

End Sub

Private Sub Добавить\_паспорт\_Click(sender As Object, e As EventArgs) Handles Добавить\_паспорт.Click

If id\_животного\_доб.Text <> "" And номер\_родословной\_доб.Text <> "" And кличка\_животного\_доб.Text <> "" And группа\_животного\_доб.Text <> "" And окрас\_животного\_доб.Text <> "" And пол\_животного\_доб.Text <> "" And имя\_мамы\_доб.Text <> "" And имя\_папы\_доб.Text <> "" And номер\_чипа\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Паспорт\_животного (ID\_Животного,Владелец,Номер\_родословной,Кличка,Порода,Дата\_рождения,Группа,Окрас,Пол,Имя\_мамы,Имя\_папы,Номер\_чипа,Дата\_имплантации,Место\_размещения) VALUES(@id\_zij,@vlad,@nomer\_rodoslov,@klichka,@poroda,@data\_roz,@gryppa,@okras,@pol,@im\_mam,@im\_pa,@nomer\_chip,@data\_implant,@mestorazm)"

Using cmd As New SqlCommand(query, conn)

id\_владельца\_животного\_доб.SelectedIndex += 1

id\_породы\_животного\_доб.SelectedIndex += 1

место\_размещения\_доб.SelectedIndex += 1

cmd.Parameters.AddWithValue("@id\_zij", id\_животного\_доб.Text)

cmd.Parameters.AddWithValue("@vlad", id\_владельца\_животного\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@nomer\_rodoslov", номер\_родословной\_доб.Text)

cmd.Parameters.AddWithValue("@klichka", кличка\_животного\_доб.Text)

cmd.Parameters.AddWithValue("@poroda", id\_породы\_животного\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@data\_roz", d3.Value)

cmd.Parameters.AddWithValue("@gryppa", группа\_животного\_доб.Text)

cmd.Parameters.AddWithValue("@okras", окрас\_животного\_доб.Text)

cmd.Parameters.AddWithValue("@pol", пол\_животного\_доб.Text)

cmd.Parameters.AddWithValue("@im\_mam", имя\_мамы\_доб.Text)

cmd.Parameters.AddWithValue("@im\_pa", имя\_папы\_доб.Text)

cmd.Parameters.AddWithValue("@nomer\_chip", номер\_чипа\_доб.Text)

cmd.Parameters.AddWithValue("@data\_implant", d4.Value)

cmd.Parameters.AddWithValue("@mestorazm", место\_размещения\_доб.SelectedIndex)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_паспорт\_животного\_актул.DataSource = Getdatalist7() : id\_животного\_доб.Clear() : номер\_родословной\_доб.Clear() : кличка\_животного\_доб.Clear() : группа\_животного\_доб.Clear() : окрас\_животного\_доб.Clear() : пол\_животного\_доб.Clear() : имя\_мамы\_доб.Clear() : имя\_папы\_доб.Clear() : номер\_чипа\_доб.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Удалить\_паспорт\_Click(sender As Object, e As EventArgs) Handles Удалить\_паспорт.Click

If кличка\_животного\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Паспорт\_животного where Кличка = @klich"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@klich", кличка\_животного\_дел.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_паспорт\_животного\_актул.DataSource = Getdatalist7() : кличка\_животного\_дел.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Изменить\_паспорт\_Click(sender As Object, e As EventArgs) Handles Изменить\_паспорт.Click

If id\_животного\_обнв.Text <> "" And номер\_родословной\_обнв.Text <> "" And имя\_животного\_обнв.Text <> "" And группа\_животного\_обнв.Text <> "" And окрас\_животного\_обнв.Text <> "" And пол\_животного\_обнв.Text <> "" And имя\_мамы\_обнв.Text <> "" And имя\_папы\_обнв.Text <> "" And номер\_чипа\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Паспорт\_животного set Владелец=@vlad\_u,Номер\_родословной=@nomer\_rodoslov\_u,Кличка=@klichka\_u,Порода=@poroda\_u,Дата\_рождения=@data\_roz\_u,Группа=@gryppa\_u,Окрас=@okras\_u,Пол=@pol\_u,Имя\_мамы=@im\_mam\_u,Имя\_папы=@im\_pa\_u,Номер\_чипа=@nomer\_chip\_u,Дата\_имплантации=@data\_implant\_u,Место\_размещения=@mestorazm\_u where ID\_Животного=@id\_zij\_u"

Using cmd As New SqlCommand(query, conn)

id\_владельца\_животного\_обнв.SelectedIndex += 1

id\_породы\_животного\_обнв.SelectedIndex += 1

место\_размещения\_обнв.SelectedIndex += 1

cmd.Parameters.AddWithValue("@id\_zij\_u", id\_животного\_обнв.Text)

cmd.Parameters.AddWithValue("@vlad\_u", id\_владельца\_животного\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@nomer\_rodoslov\_u", номер\_родословной\_обнв.Text)

cmd.Parameters.AddWithValue("@klichka\_u", имя\_животного\_обнв.Text)

cmd.Parameters.AddWithValue("@poroda\_u", id\_породы\_животного\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@data\_roz\_u", d5.Value)

cmd.Parameters.AddWithValue("@gryppa\_u", группа\_животного\_обнв.Text)

cmd.Parameters.AddWithValue("@okras\_u", окрас\_животного\_обнв.Text)

cmd.Parameters.AddWithValue("@pol\_u", пол\_животного\_обнв.Text)

cmd.Parameters.AddWithValue("@im\_mam\_u", имя\_мамы\_обнв.Text)

cmd.Parameters.AddWithValue("@im\_pa\_u", имя\_папы\_обнв.Text)

cmd.Parameters.AddWithValue("@nomer\_chip\_u", номер\_чипа\_обнв.Text)

cmd.Parameters.AddWithValue("@data\_implant\_u", d6.Value)

cmd.Parameters.AddWithValue("@mestorazm\_u", место\_размещения\_обнв.SelectedIndex)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_паспорт\_животного\_актул.DataSource = Getdatalist7() : id\_животного\_обнв.Clear() : номер\_родословной\_обнв.Clear() : имя\_животного\_обнв.Clear() : группа\_животного\_обнв.Clear() : окрас\_животного\_обнв.Clear() : пол\_животного\_обнв.Clear() : имя\_мамы\_обнв.Clear() : имя\_папы\_обнв.Clear() : номер\_чипа\_обнв.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Function Getdatalist7() As DataTable

Dim dt As New DataTable

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("select Паспорт\_животного.ID\_Животного as 'Номер животного',Владельцы.ФИО, Паспорт\_животного.Номер\_родословной as 'Номер родословной',Паспорт\_животного.Кличка,Породы.Название As'Порода',Паспорт\_животного.Дата\_рождения as 'Дата рождения',Паспорт\_животного.Группа,Паспорт\_животного.Окрас,Паспорт\_животного.Пол,Паспорт\_животного.Имя\_мамы as 'Имя мамы',Паспорт\_животного.Имя\_папы as 'Имя папы',Паспорт\_животного.Номер\_чипа as 'Номер чипа' ,Паспорт\_животного.Дата\_имплантации as 'Дата имплантации',Место\_размещения\_электронного\_чипа.Где from Паспорт\_животного join Владельцы on Паспорт\_животного.Владелец=Владельцы.ID\_Владельца join Породы on Паспорт\_животного.Порода= Породы.ID\_Породы join Место\_размещения\_электронного\_чипа on Паспорт\_животного.Место\_размещения= Место\_размещения\_электронного\_чипа.ID\_Место", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader)

End Using

End Using

Return dt

End Function

Private Sub ComboBox2\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles получ\_награды\_обнв.SelectedIndexChanged

End Sub

Private Sub Добавить\_получ\_награды\_Click(sender As Object, e As EventArgs) Handles Добавить\_получ\_награды.Click

If id\_получ\_награды\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Получение\_награды (ID\_Получ,Кличка\_животного,Наименование\_награды,Дата\_получения,Прошлая\_дата\_получения) VALUES(@idPolych,@klichJiv,@imNagrd,@dataPolych,@Proshdata)"

Using cmd As New SqlCommand(query, conn)

ID\_клички\_доб.SelectedIndex += 1

получ\_награды\_доб.SelectedIndex += 1

cmd.Parameters.AddWithValue("@idPolych", id\_получ\_награды\_доб.Text)

cmd.Parameters.AddWithValue("@klichJiv", ID\_клички\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@imNagrd", получ\_награды\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@dataPolych", d7.Value)

cmd.Parameters.AddWithValue("@Proshdata", d8.Value)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_получ\_награды\_актул.DataSource = Getdatalist8() : id\_получ\_награды\_доб.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Удалить\_получ\_награды\_Click(sender As Object, e As EventArgs) Handles Удалить\_получ\_награды.Click

If id\_получ\_награды\_уд.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Получение\_награды where ID\_Получ = @iddlete"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@iddlete", id\_получ\_награды\_уд.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_получ\_награды\_актул.DataSource = Getdatalist8() : id\_получ\_награды\_уд.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Изменить\_получ\_награды\_Click(sender As Object, e As EventArgs) Handles Изменить\_получ\_награды.Click

If id\_получ\_награды\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Получение\_награды set Кличка\_животного =@ima\_u ,Наименование\_награды=@nameNagrad\_u,Дата\_получения=@data\_polych\_u,Прошлая\_дата\_получения=@proshla\_\_u where ID\_Получ =@id\_polych\_u"

Using cmd As New SqlCommand(query, conn)

ID\_клички\_обнв.SelectedIndex += 1

получ\_награды\_обнв.SelectedIndex += 1

cmd.Parameters.AddWithValue("@id\_polych\_u", id\_получ\_награды\_обнв.Text)

cmd.Parameters.AddWithValue("@ima\_u", ID\_клички\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@nameNagrad\_u", получ\_награды\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@data\_polych\_u", d9.Value)

cmd.Parameters.AddWithValue("@proshla\_\_u", d10.Value)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_получ\_награды\_актул.DataSource = Getdatalist8() : id\_получ\_награды\_обнв.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Function Getdatalist8() As DataTable

Dim dt As New DataTable

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("Select Получение\_награды.ID\_Получ as 'Номер получения', Паспорт\_животного.Кличка, Награды.Наименование, Получение\_награды.Дата\_получения as 'Дата получения',Получение\_награды.Прошлая\_дата\_получения as 'Прошлая дата получения' from Получение\_награды join Паспорт\_животного on Получение\_награды.Кличка\_животного=Паспорт\_животного.ID\_Животного join Награды on Получение\_награды.Наименование\_награды= Награды.ID\_Награды", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader)

End Using

End Using

Return dt

End Function

Private Sub Добавить\_вет\_свед\_Click(sender As Object, e As EventArgs) Handles Добавить\_вет\_свед.Click

If id\_вет\_св\_доб.Text <> "" And имя\_картинки\_доб.Text <> "" And формат\_кратинки\_доб.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "INSERT INTO Ветеринарное\_свидетельство (ID\_Вет,Имя\_животного,Наименование\_прививки,Привика\_от,Дата\_принятия,Срок\_действия\_прививки,Имя\_картинки,Формат\_картинки) VALUES(@idvet,@nameJiv,@namePriv,@privot,@dataprin,@srok,@nameimage,@format)"

Using cmd As New SqlCommand(query, conn)

id\_имя\_жив\_вет\_доб.SelectedIndex += 1

id\_наименов\_прирв\_вет\_доб.SelectedIndex += 1

id\_прив\_от\_вет\_доб.SelectedIndex += 1

cmd.Parameters.AddWithValue("@idvet", id\_вет\_св\_доб.Text)

cmd.Parameters.AddWithValue("@nameJiv", id\_имя\_жив\_вет\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@namePriv", id\_наименов\_прирв\_вет\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@privot", id\_прив\_от\_вет\_доб.SelectedIndex)

cmd.Parameters.AddWithValue("@dataprin", d11.Value)

cmd.Parameters.AddWithValue("@srok", d12.Value)

cmd.Parameters.AddWithValue("@nameimage", имя\_картинки\_доб.Text)

cmd.Parameters.AddWithValue("@format", формат\_кратинки\_доб.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_ветеринарное\_свидетельство.DataSource = Getdatalist9() : имя\_картинки\_доб.Clear() : формат\_кратинки\_доб.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Удалить\_вет\_свед\_Click(sender As Object, e As EventArgs) Handles Удалить\_вет\_свед.Click

If id\_вет\_св\_дел.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "delete from Ветеринарное\_свидетельство where ID\_Вет = @idVet\_del"

Using cmd As New SqlCommand(query, conn)

cmd.Parameters.AddWithValue("@idVet\_del", id\_вет\_св\_дел.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_ветеринарное\_свидетельство.DataSource = Getdatalist9() : id\_вет\_св\_дел.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Sub Изменить\_вет\_свед\_Click(sender As Object, e As EventArgs) Handles Изменить\_вет\_свед.Click

If id\_вет\_св\_обнв.Text <> "" And имя\_картинки\_обнв.Text <> "" And формат\_кратинки\_обнв.Text <> "" Then

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Dim query = "update Ветеринарное\_свидетельство set Имя\_животного=@nameJiv\_up ,Наименование\_прививки=@namePriv\_up,Привика\_от=@privot\_up,Дата\_принятия=@dataprin\_up ,Срок\_действия\_прививки=@srok\_up,Имя\_картинки=@nameimage\_up,Формат\_картинки=@format\_up where ID\_Вет=@idvet\_up"

Using cmd As New SqlCommand(query, conn)

id\_имя\_жив\_вет\_обнв.SelectedIndex += 1

id\_наименов\_прирв\_вет\_обнв.SelectedIndex += 1

id\_прив\_от\_вет\_обнв.SelectedIndex += 1

cmd.Parameters.AddWithValue("@idvet\_up", id\_вет\_св\_обнв.Text)

cmd.Parameters.AddWithValue("@nameJiv\_up", id\_имя\_жив\_вет\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@namePriv\_up", id\_наименов\_прирв\_вет\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@privot\_up", id\_прив\_от\_вет\_обнв.SelectedIndex)

cmd.Parameters.AddWithValue("@dataprin\_up", d13.Value)

cmd.Parameters.AddWithValue("@srok\_up", d14.Value)

cmd.Parameters.AddWithValue("@nameimage\_up", имя\_картинки\_обнв.Text)

cmd.Parameters.AddWithValue("@format\_up", формат\_кратинки\_обнв.Text)

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

End Using

End Using

DGV\_ветеринарное\_свидетельство.DataSource = Getdatalist9() : имя\_картинки\_обнв.Clear() : формат\_кратинки\_обнв.Clear()

Else

MessageBox.Show("Заполните поле(я) данными")

End If

End Sub

Private Function Getdatalist9() As DataTable

Dim dt As New DataTable

Dim connString As String = ConfigurationManager.ConnectionStrings("ZelZooExpo.My.MySettings.animalBDConnectionString1").ConnectionString

Using conn As New SqlConnection(connString)

Using cmd As New SqlCommand("Select Ветеринарное\_свидетельство.ID\_Вет as 'Номер вет', Паспорт\_животного.Кличка,Прививка.Название\_прививки as 'Название прививки',ОтчегоПрививка.Прививка\_от as 'Прививка от',Ветеринарное\_свидетельство.Дата\_принятия as 'Дата получения',Ветеринарное\_свидетельство.Срок\_действия\_прививки as 'Срок действия прививки',Ветеринарное\_свидетельство.Имя\_картинки as 'Имя картинки',Ветеринарное\_свидетельство.Формат\_картинки as 'Формат картинки' from Ветеринарное\_свидетельство join Паспорт\_животного on Ветеринарное\_свидетельство.Имя\_животного=Паспорт\_животного.ID\_Животного join Прививка on Ветеринарное\_свидетельство.Наименование\_прививки=Прививка.ID\_прив join ОтчегоПрививка on Ветеринарное\_свидетельство.Привика\_от=ОтчегоПрививка.ID\_отЧего", conn)

conn.Open()

Dim reader As SqlDataReader = cmd.ExecuteReader()

dt.Load(reader)

End Using

End Using

Return dt

End Function

Private Sub id\_страны\_доб\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles id\_страны\_доб.KeyPress, id\_страны\_обнв.KeyPress, номер\_чипа\_обнв.KeyPress, номер\_чипа\_доб.KeyPress, номер\_родословной\_обнв.KeyPress, номер\_родословной\_доб.KeyPress, номер\_дома\_обнв.KeyPress, номер\_дома\_доб.KeyPress, id\_прививки\_обнв.KeyPress, id\_прививки\_доб.KeyPress, id\_породы\_обнв.KeyPress, id\_породы\_доб.KeyPress, id\_получ\_награды\_уд.KeyPress, id\_получ\_награды\_обнв.KeyPress, id\_получ\_награды\_доб.KeyPress, id\_отчего\_прив\_обнв.KeyPress, id\_отчего\_прив\_доб.KeyPress, id\_награды\_обнв.KeyPress, id\_награды\_доб.KeyPress, id\_места\_обнв.KeyPress, id\_места\_доб.KeyPress, id\_животного\_обнв.KeyPress, id\_животного\_доб.KeyPress, id\_владельца\_обнв.KeyPress, id\_владельца\_доб.KeyPress, id\_вет\_св\_обнв.KeyPress, id\_вет\_св\_доб.KeyPress, id\_вет\_св\_дел.KeyPress, колживнаг\_обнов.KeyPress, колживнаг\_доб.KeyPress, колвоприв\_обнов.KeyPress, колвоприв\_доб.KeyPress

If Not IsNumeric(e.KeyChar) AndAlso Asc(e.KeyChar) <> 8 Then e.Handled = True

End Sub

Private Sub название\_страны\_доб\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles название\_страны\_доб.KeyPress, название\_страны\_удл.KeyPress, название\_страны\_обнв.KeyPress, формат\_кратинки\_обнв.KeyPress, формат\_кратинки\_доб.KeyPress, ФИО\_обнв.KeyPress, ФИО\_доб.KeyPress, ФИО\_дел.KeyPress, улица\_обнв.KeyPress, улица\_доб.KeyPress, страна\_производитель\_обнв.KeyPress, страна\_производитель\_доб.KeyPress, прививка\_от\_обнв.KeyPress, прививка\_от\_доб.KeyPress, прививка\_от\_дел.KeyPress, пол\_животного\_обнв.KeyPress, пол\_животного\_доб.KeyPress, окрас\_животного\_обнв.KeyPress, окрас\_животного\_доб.KeyPress, наименование\_прививки\_обнв.KeyPress, наименование\_прививки\_доб.KeyPress, наименование\_прививки\_дел.KeyPress, наименование\_награды\_обнв.KeyPress, наименование\_награды\_доб.KeyPress, наименование\_награды\_дел.KeyPress, название\_породы\_обнв.KeyPress, название\_породы\_доб.KeyPress, название\_породы\_дел.KeyPress, кличка\_животного\_доб.KeyPress, имя\_папы\_обнв.KeyPress, имя\_папы\_доб.KeyPress, имя\_мамы\_обнв.KeyPress, имя\_мамы\_доб.KeyPress, имя\_животного\_обнв.KeyPress, группа\_животного\_обнв.KeyPress, группа\_животного\_доб.KeyPress, город\_обнв.KeyPress, город\_доб.KeyPress, где\_обнв.KeyPress, где\_доб.KeyPress, где\_дел.KeyPress, кличка\_животного\_дел.KeyPress

Dim chr As Char = e.KeyChar

Dim iCh As Integer = AscW(chr)

If Not Char.IsLetter(chr) AndAlso (chr <> vbBack And chr <> ChrW(32) And iCh + iCh <> 52 And iCh + iCh <> 48 \_

And iCh + iCh <> 44 And iCh + iCh <> 2 And iCh + iCh <> 6) Then e.Handled = True

End Sub

Private Sub aktyalizat\_KeyDown(sender As Object, e As KeyEventArgs) Handles MyBase.KeyDown

If e.KeyCode = Keys.F1 Then Help.ShowHelp(Me, "ZelZooExpo\_Info.chm")

End Sub

Private Sub id\_страны\_влад\_доб\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles id\_страны\_влад\_доб.SelectedIndexChanged

End Sub

End Class