ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК

КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ

**Лабораторная работа №6**

**по дисциплине «Информационные системы в экономике и управлении»**

на тему: «Проектирование информационного обеспечения»

**Выполнила:**

Студентка гр.ПИ-185-1

Ретунский К.В.

**Проверил:**

Доцент кафедры

информационных систем,

к.т.н., доцент

Карякин Ю.Е.

Тюмень 2019

Оглавление

[Средства разработки 3](#_Toc36175097)

[Руководство пользователя 6](#_Toc36175098)

[Заключение 8](#_Toc36175099)

[Список литературы 9](#_Toc36175100)

# Средства разработки

Для разработки проекта, описанного в предыдущих практических работах, была выбрана среда Visual Studio .NET и язык программирования C#.

Операционная среда разработки MS Visual Studio на сегодняшний день является предпочтительным выбором многих разработчиков, работающих на платформе Windows. Среда позволяет эффективно создавать сложные приложения в течение короткого периода времени. Модель данной среды существенно богата и использует такие понятия как решение (solution), проект, пространство имен (namespace) и сборка (assembly). Файл проекта содержит перечисление исходных файлов и прочих ресурсов, из которых система будет строить приложение. В решение среды Visual Studio входят несколько проектов, которые могут быть зависимыми или независимыми друг от друга.

C# — один из самых популярных языков программирования. Он достаточно прост для понимания, что является главным преимуществом для начинающих/неопытных разработчиков. Также данный язык программирования достаточно гибок. Разнообразие приложений, которые могут быть разработаны с помощью С#, .Net и Visual Studio практически безгранично.

Конечно, все это возможно выполнять и с помощью других языков программирования, но обычно, в таких случаях, используются сторонние инструменты других разработчиков. Программисты, которые работают с C#, имеют сплоченный набор инструментов, поддерживаемых Microsoft для разработки любого типа приложения.

Хранение данных обеспечивается с помощью Microsoft Access. Access является надежной системой управления реляционными базами данных

Достоинства Microsoft Access:

* очень простой графический интерфейс, который позволяет не только создавать собственную базу данных, но и разрабатывать приложения, используя встроенные средства;
* хранит все данные в одном файле, хотя и распределяет их по разным таблицам, как и положено реляционной СУБД. К этим данным относится не только информация в таблицах, но и другие объекты базы данных;
* предлагает большое количество Мастеров, которые выполняют основную работу за пользователя при работе с данными и разработке приложений, помогают избежать рутинных действий и облегчают работу неискушенному в программировании пользователю;
* постоянно обновляется производителем, поддерживает множество языков

Вся база данных хранится в Microsoft Access.

Для примера структур таблиц продемонстрируем структуру таблицы «Персонал» (рисунок 1) и данные, которые в ней хранятся (рисунок 2):

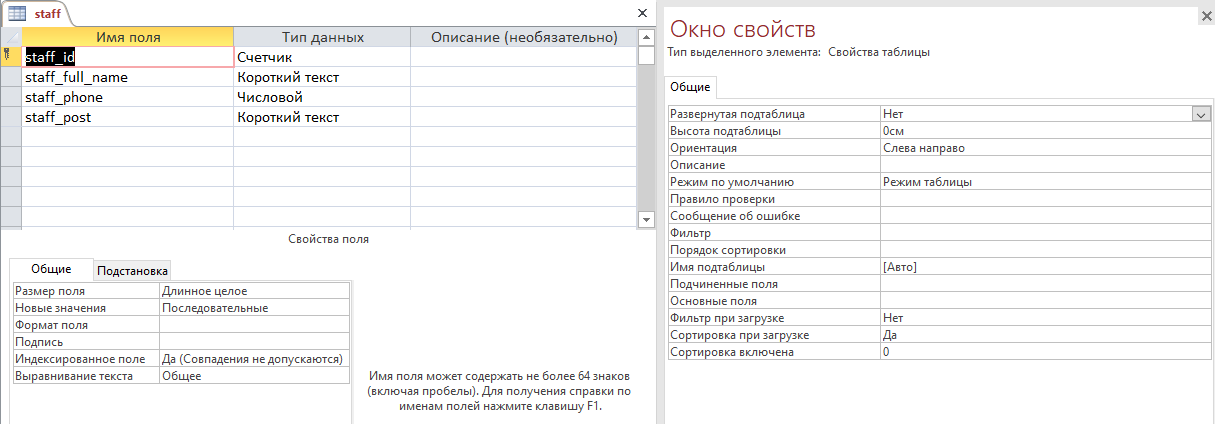


Рисунок 1 – структура таблицы «Персонал»

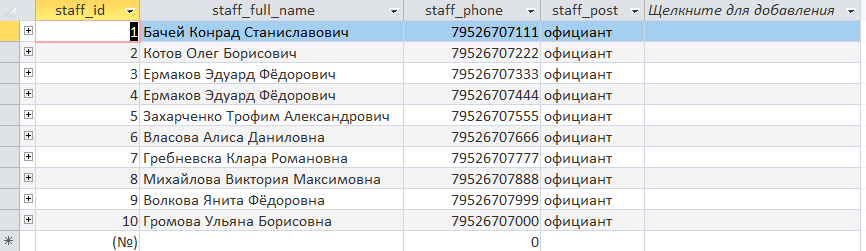


Рисунок 2 – данные таблицы «Персонал»

Связывание БД и нашего приложения происходит с помощью интерфейса Visual Studio, в котором мы указываем, с каким типом базы данных мы хотим работать, имя сервера, имя базы данных, а так же таблицы, которые мы хотим связать. Продемонстрируем форму подключения базы данных к нашему проекту, на которой интуитивно понимается подключение базы данных (рисунок 3):

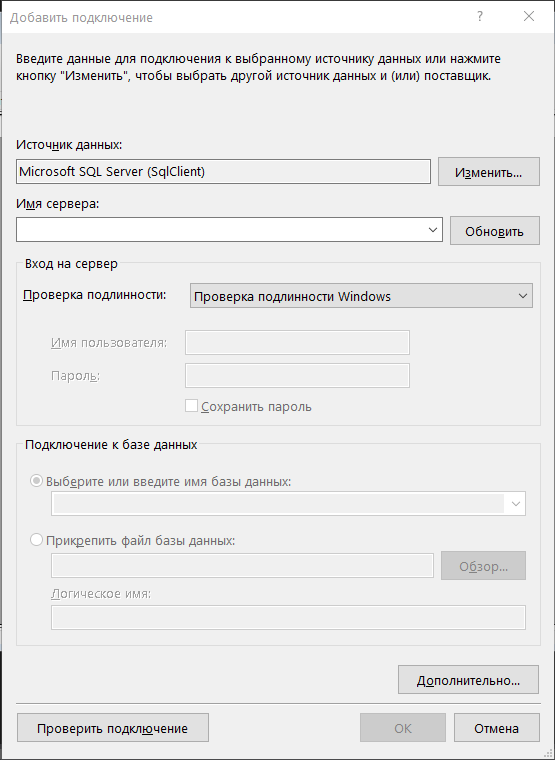


Рисунок 3 – Демонстрация формы добавление базы данных

# Руководство пользователя

Главная форма имеет следующий вид (рисунок 4):

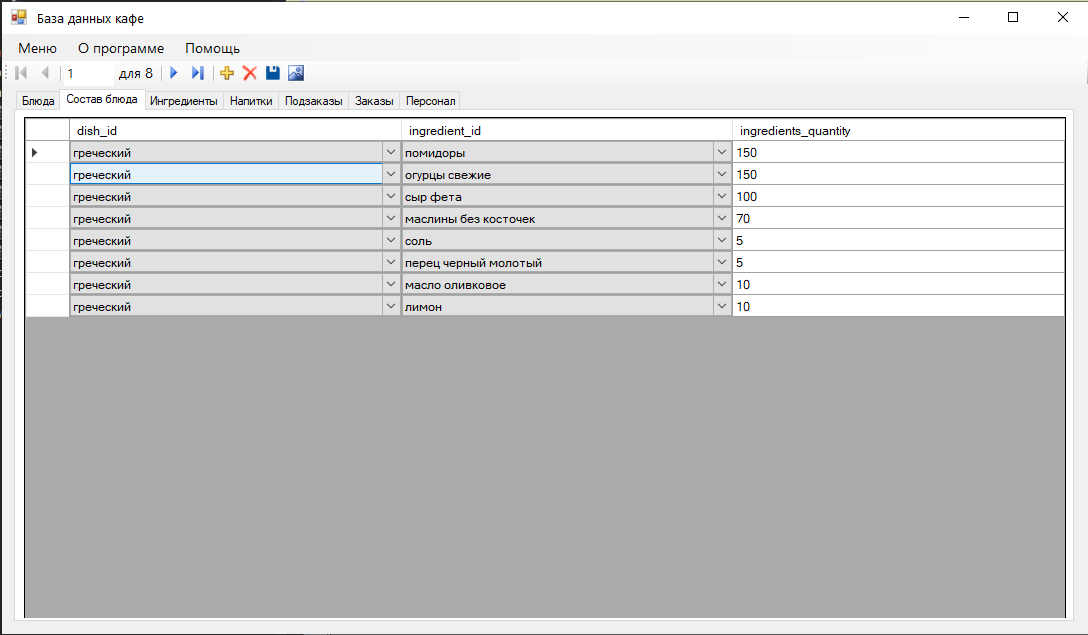


Рисунок 4 – демонстрация главной формы

Верхнее меню содержит три пункта: «Меню», «Помощь» и «О программе». Раздел «О программе» предназначен для получения информации о приложении и разработчике.

Раздел «Помощь» предназначен для тех, кто не смог разобраться в программном интерфейсе и немного направит пользователя.

Раздел «Меню» имеет весь функционал, который присутствует в панели BindingNavigator, для тех, кому привычнее использовать кнопки с надписями.

Так же пользователю не придется искать в какой строке находится название определенного ключа. Потому-что это изменено посредствам DataGridVeiw, где можно изменить ключ на нормальное название. Для удобства так же имеется выпадающий список, в котором пользователь может выбрать уже известные ему данные. Пример выпадающего списка приведен на рисунке 5.

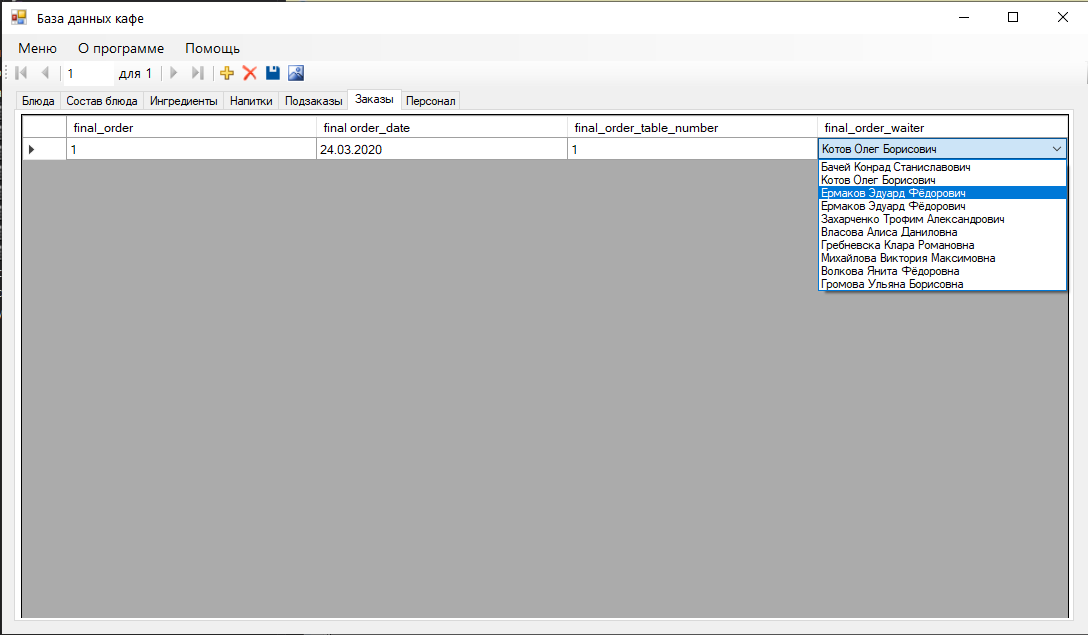


Рисунок 5 - Пример выпадающего списка

# Заключение

В ходе данной практической работы были рассмотрены средства разработки программы, с помощью которой выполняется заполнение базы данных кафе.

Целью проектирования являлось определение внутренних свойств системы и детализации её видимых свойств на основе выданных «заказчиком» требований к ПО (исходные условия задачи). Эти требования были подвергнуты анализу.

Ранее для выражения характеристик системы были использованы такие нотации, как DFD, IDEF0, а также была построена логическая и концептуальная информационные модели.

На этапе проектирования были выработаны проектные решения, связанные с выбором платформ и технологий, на основе которых будет функционировать система, языка (или комбинации) языков реализации, определены требования к пользовательскому интерфейсу, выбрана наиболее подходящая СУБД и т.д.

# Список литературы

1. Синтаксис строки подключения — https://docs.microsoft.com/ru-ru/dotnet/framework/data/adonet/connection-string-syntax

2. Преимущества и недостатки C# — https://shwanoff.ru/plus-minus-c-sharp/

3. Язык программирования C# — https://learn-code.ru/yazyki-programmirovaniya/c-sharp

4. Проектирование программного обеспечения — https://habr.com/ru/company/edison/blog/267569/