|  |
| --- |
| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| **федеральное государственное бюджетное образовательное учреждение высшего образования «Российский экономический университет имени Г.В. Плеханова»**  **Московский приборостроительный техникум** |

**ОТЧЕТ**

по учебной практике

УП 01.01 «Прикладное программирование»

*индекс по УП и наименование практики*

Профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем»

*индекс по УП и наименование профессионального модуля*

Специальность 09.02.07 «Информационные системы и программирование»

*код и наименование специальности*

Студент Николаева Александра Николаевна

*(фамилия, имя, отчество)*

Группа П50-2-22

Руководитель по практической подготовке от техникума

Дмитрий Владимирович Мысев

*(фамилия, имя, отчество)*

«29» марта 2023 года

Лабораторные работы

[ЛАБОРАТОРНАЯ РАБОТА №1 3](#_Toc163018731)

[«Подключение базы данных. Чтение данных из БД» 3](#_Toc163018732)

[ЛАБОРАТОРНАЯ РАБОТА №2 20](#_Toc163018733)

[«Добавление, изменение, удаление данных в БД» 20](#_Toc163018734)

[ЛАБОРАТОРНАЯ РАБОТА №3 21](#_Toc163018735)

[«Отображение нескольких таблиц из БД» 21](#_Toc163018736)

# ЛАБОРАТОРНАЯ РАБОТА №1

## «Подключение базы данных. Чтение данных из БД»

**Цель работы:** создать приложение WPF, которое подключается к нормализованной базе данных с 2-3 связанными таблицами, отображая данные в отдельных окнах или страницах, используя подходы Entity Framework и DataSet.

**Ход работы:**

* **Настройки и подключения**

Сперва создаем базу данным в SQL на тему «Магазин косметики», состоящую из 3 таблиц: Категории, Страны и Продукты. А после заполняем эти таблицы данными:

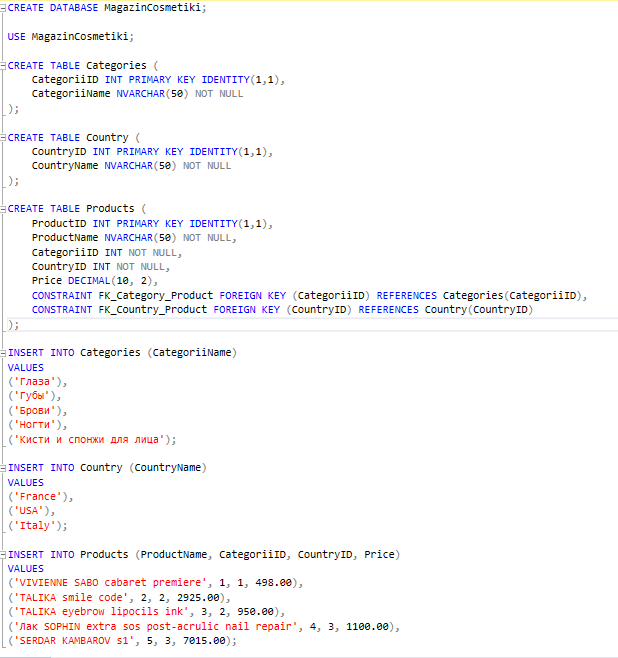


Рисунок 1- База данных "Магазин косметики"

Далее создаем проект «Приложение WPF (.NET Framework)».

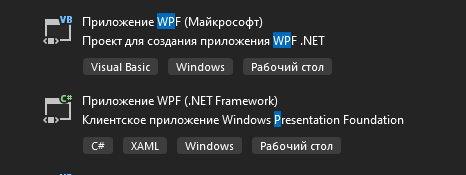


Рисунок 2- Приложение WPF (.NET Framework)

Для того, чтобы подключиться к БД, нужно в Visual Studio выбрать «Средства» -> «Подключиться к базе данных». Перед нами появится окно «Выбор источник данных». Здесь выбираем «Microsoft SQL Server». Поставщиком данных будет .NET Framework:

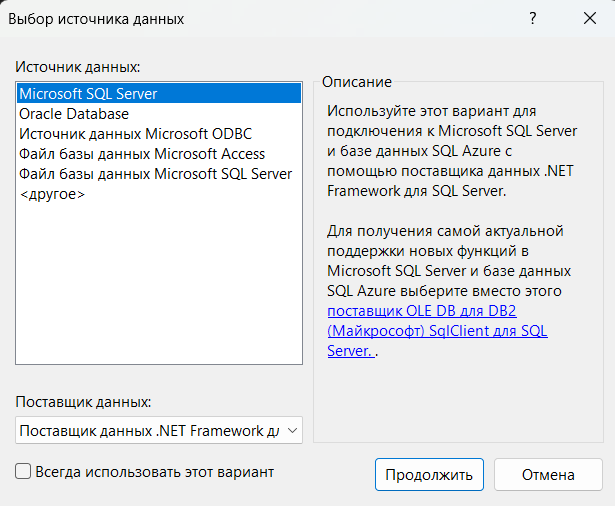


Рисунок 3- Выбор источника данных

В появившемся окне указываем имя сервера и проверку подлинности. При корректном выполнении вышенаписанных пунктов, появится список с базами данных. Выбираем нашу базу данных- MagazinCosmetiki.

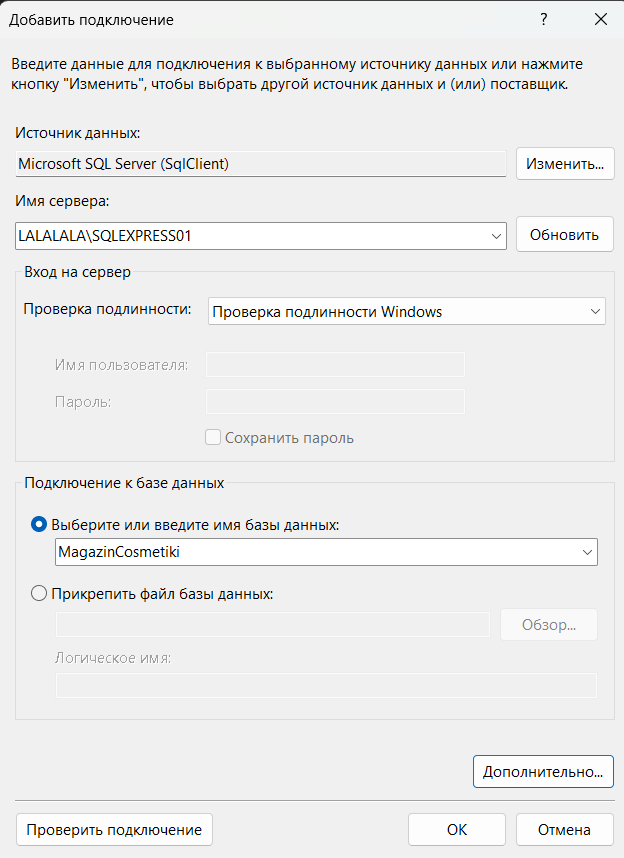


Рисунок 4- Добавление подключения

В доп.настройках поставим TrustServerCertificate = True.

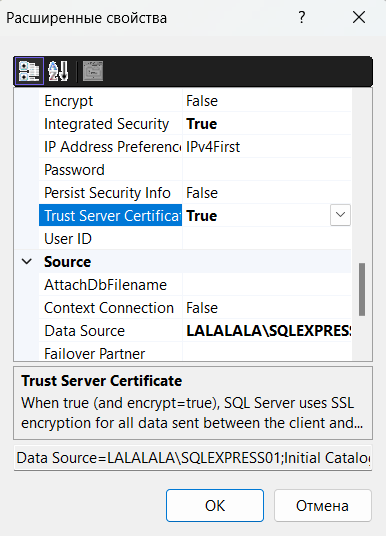


Рисунок 5- Доп.настройки

* Проект DataSet

Далее нам нужно создать объект, при помощи которого мы в коде будем работать с бд – набор данных (DataSet), подключаем его при помощи мастера. 1. «Проект» -> «Добавить новый источник данных» -> откроется мастер настройки источника данных:

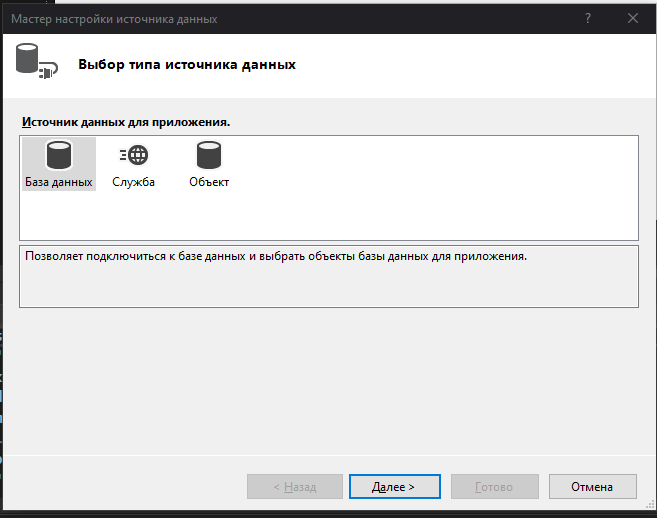


Рисунок 6- Мастер настройки источника данных

Нажимаем «Далее» и в выборе модели базы данных выбираем «Набор данных»:

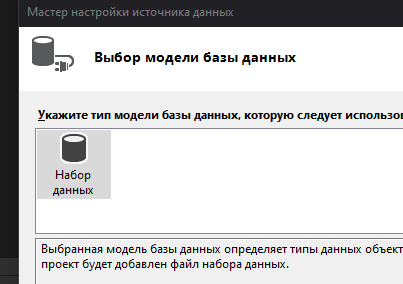


Рисунок 7- Модели бд

Далее выбираем строку подключения – заранее подключенную БД. В следующем окне оставляем все как есть.

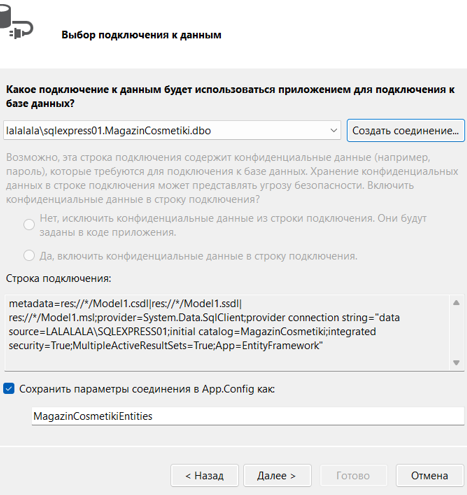


Рисунок 8- Выбор подключения к бд

При выборе объектов базы данных выбираем то, с чем мы будем работать- каждая таблица внутри БД. Нажимаем «Готово» и наш набор данных создастся.

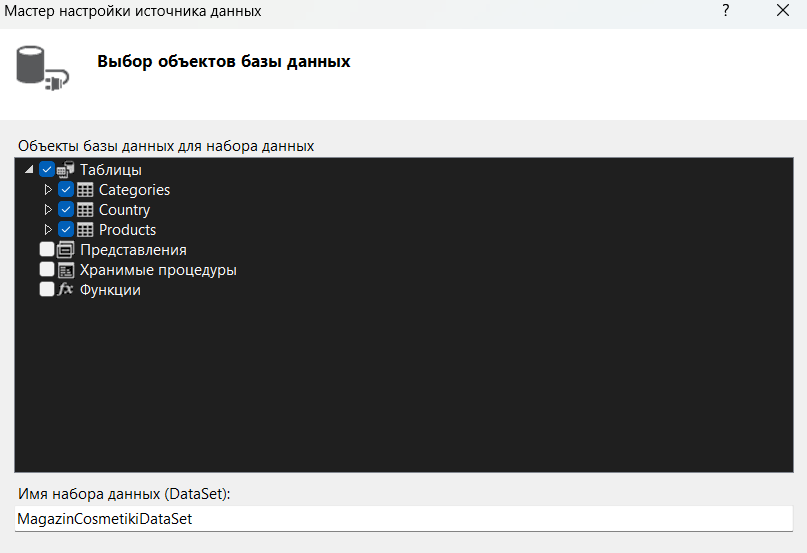


Рисунок 9- Выбор объектов

В обозревателе решений появился набор данных, который называется как MagazinCosmetikiDataSet.xsd и включает в себя 4 файла.

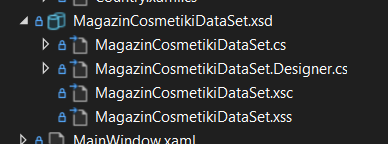


Рисунок 10- Набор данных

* Проект EF

Создаем объект, при помощи которого мы в коде будем работать с бд – Модель ADO.NET. Через ПКМ по проекту -> Добавить -> Создать элемент->Выбрать Модель ADO.NET

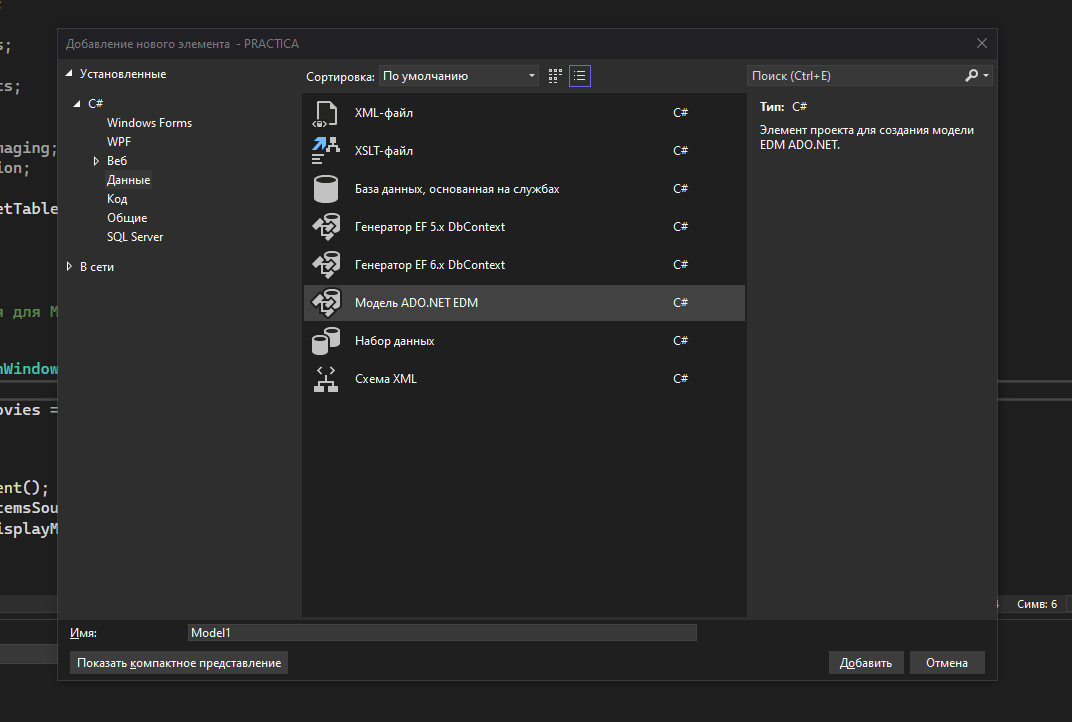


Рисунок 11- Модель ADO.NET

В появившемся окне выбираем конструктор:

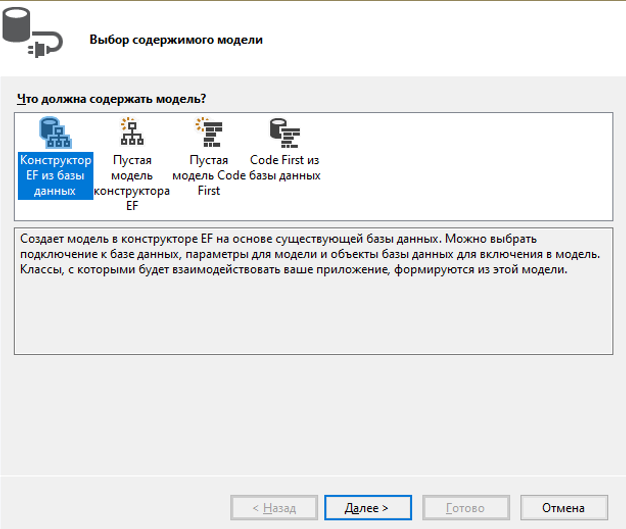


Рисунок 12- выбор содержимого модели

Выбираем подключение и нажимаем «Далее»:

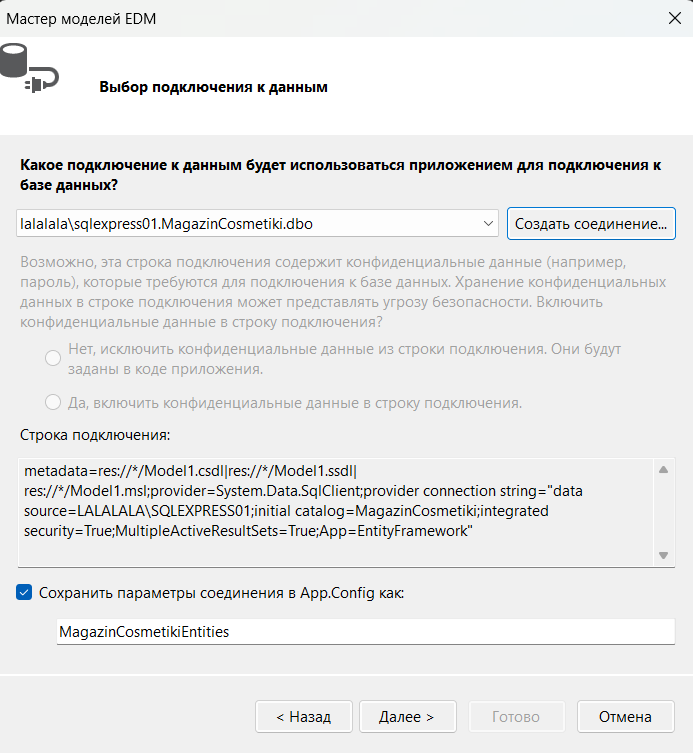


Рисунок 13- выбор подключения

Версию оставляем по умолчанию:

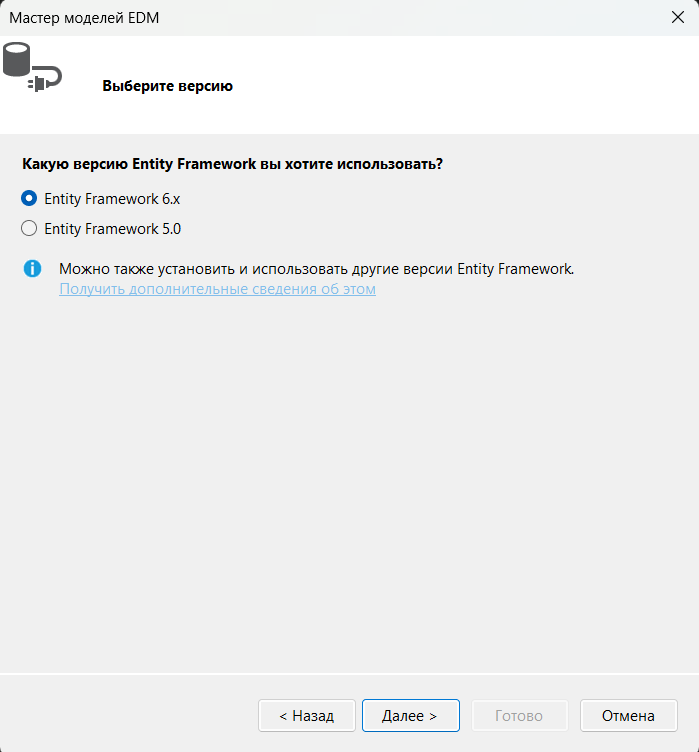


Рисунок 14- Выбор версии

В выборе параметров и объектов БД нажимаем на наши таблицы:

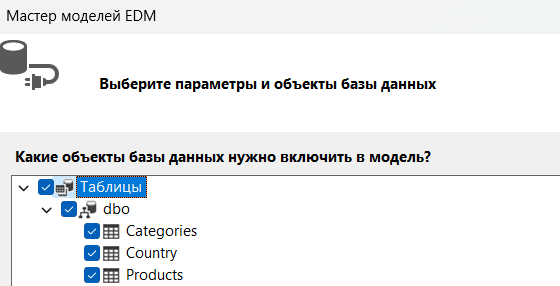


Рисунок 15- выбор объектов

Нажимая готово, он создаст модель.

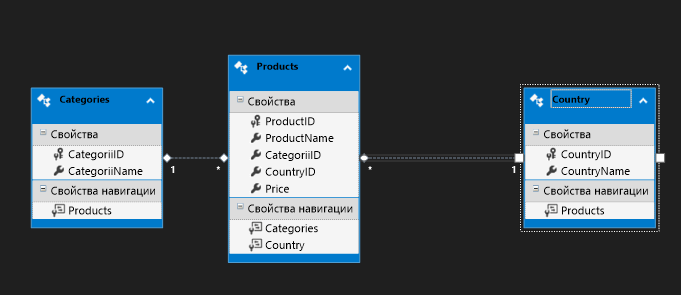


Рисунок 16- Модель

* **Проект на EntityFramework**

Создаем адаптивную верстку с тремя кнопками. При нажатии на них будет открываться страничка с выбранной таблицей:

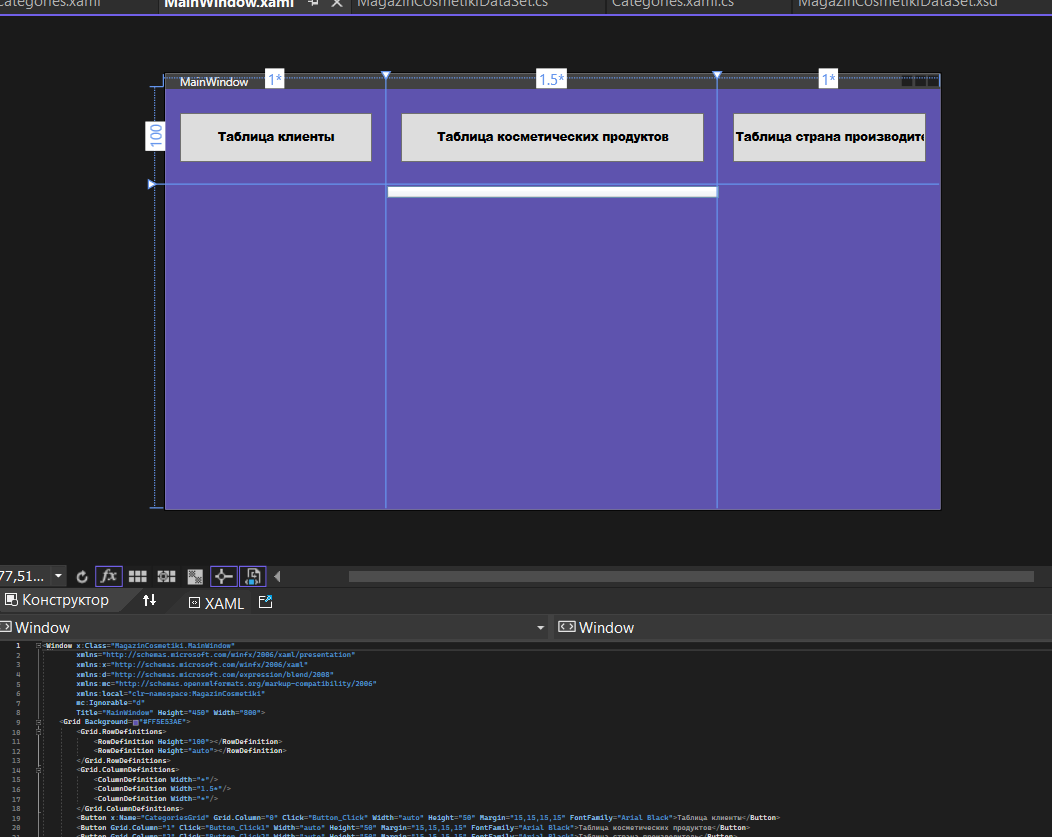


Рисунок 17- Верстка

Также создаем три страницы для просмотра таблиц:

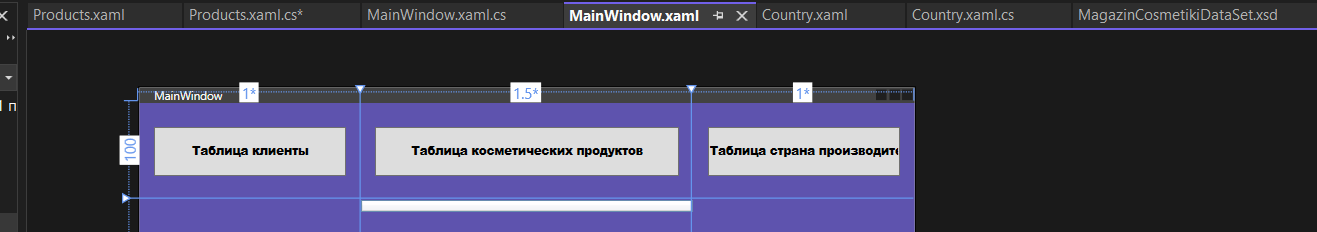


Рисунок 18- Страницы

В MainWindow.xaml.cs пишем код для кликов, чтобы при нажатии на кнопку открывалась нужная страница:

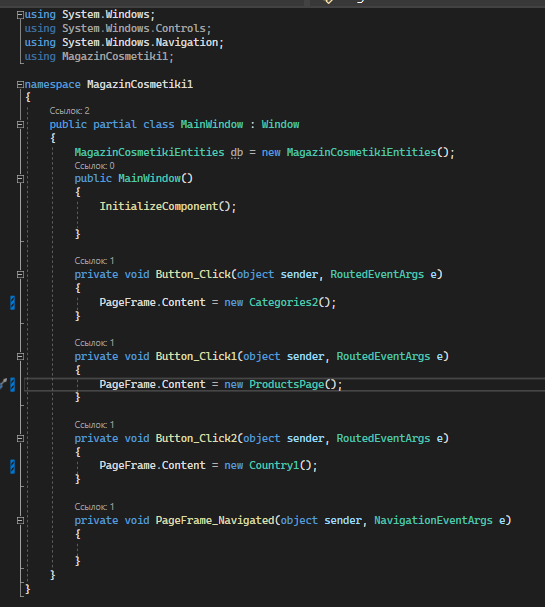


Рисунок 19- MainWindow.xaml.cs

Теперь в каждой странице выводим таблицу с данными:

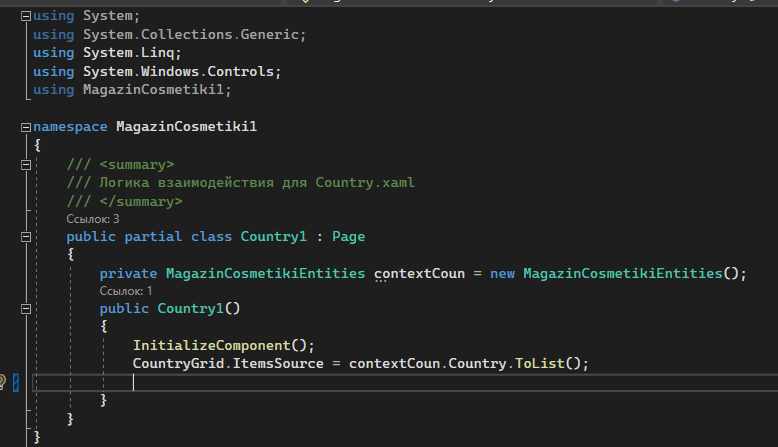


Рисунок 20- Country

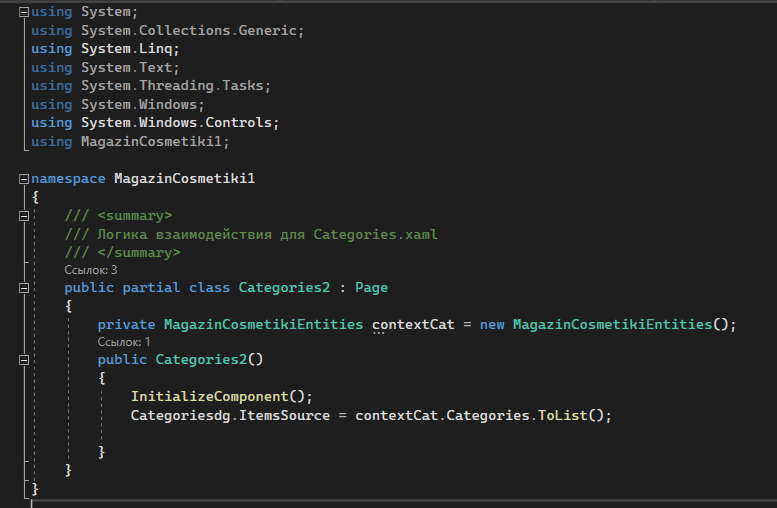


Рисунок 21- Categories2

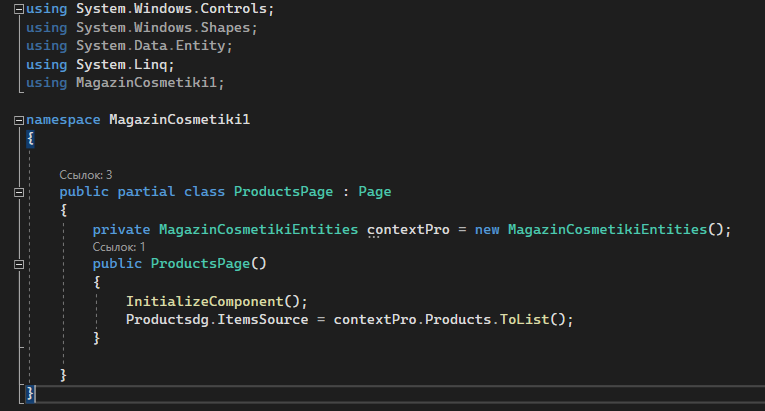


Рисунок 22- ProductsPage

Результат проекта:

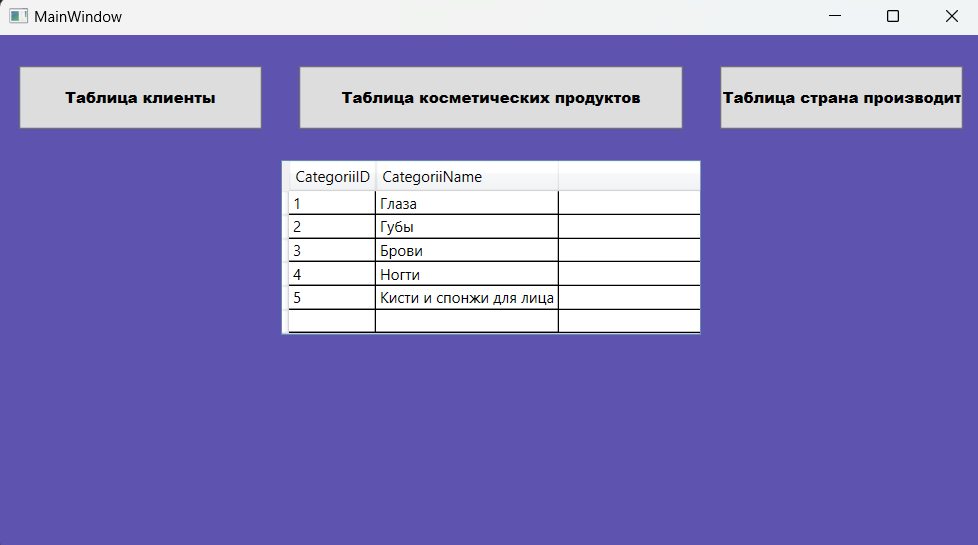


Рисунок 23 Результат

* **Проект на DataSet**

Настраиваем вывод данных таблицы на страницу в формате DataSet

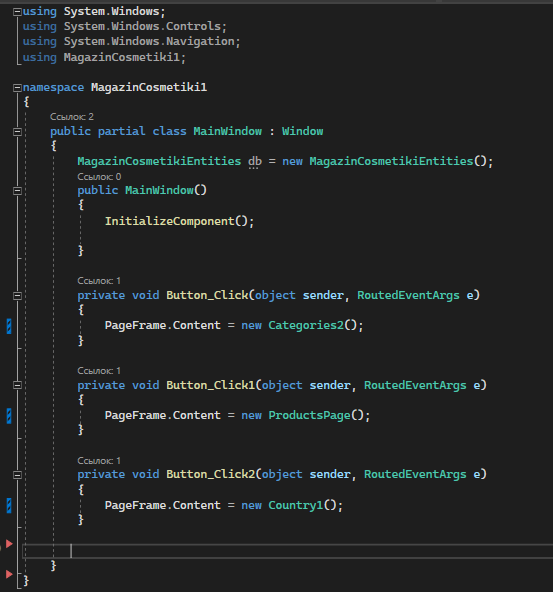


Рисунок 24- MainWindow.xaml.cs

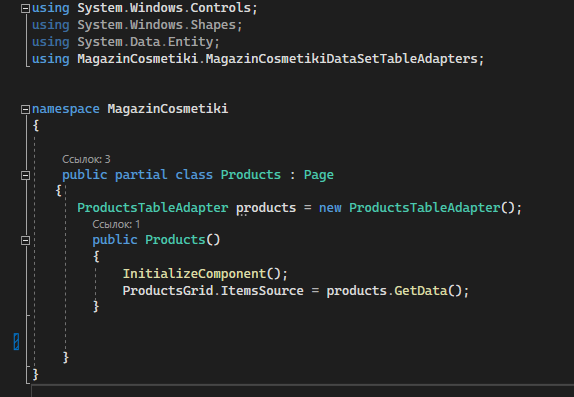


Рисунок 25- Products

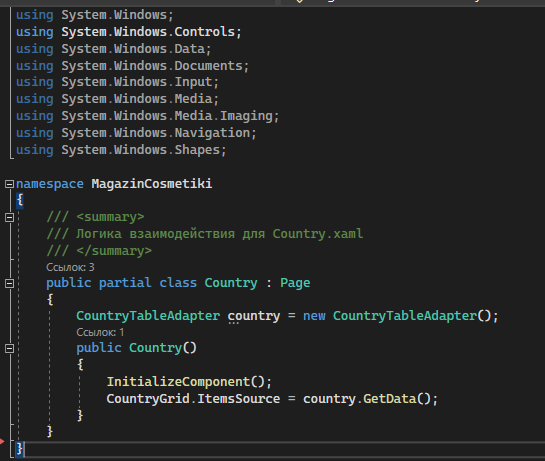


Рисунок 26- Country

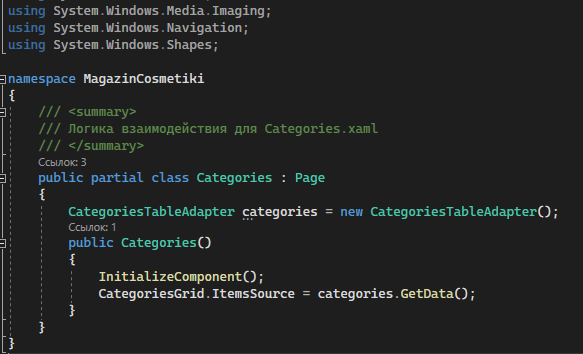


Рисунок 27- Categories

В результате при выводе таблиц будут фантомные столбцы, которые мы уберем:

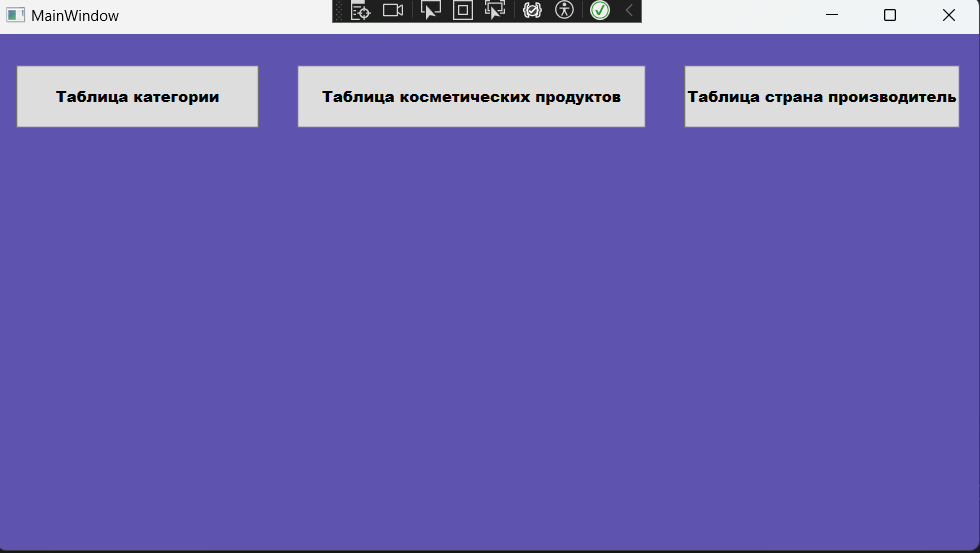


Рисунок 28- Результат

Вывод: было создано приложение WPF, которое подключается к нормализованной базе данных с 2-3 связанными таблицами, отображая данные в отдельных окнах или страницах, используя подходы Entity Framework и DataSet.

# ЛАБОРАТОРНАЯ РАБОТА №2

## «Добавление, изменение, удаление данных в БД»

Цель работы: создать приложение WPF, которое подключается к нормализованной базе данных с 2-3 связанными таблицами, отображая данные в отдельных окнах или страницах, используя подходы Entity Framework и DataSet. Добавить функции удаления, добавления и изменения данных к каждой таблице.

Вывод: Было создано приложение WPF, которое подключается к нормализованной базе данных с 2-3 связанными таблицами, отображая данные в отдельных окнах или страницах, используя подходы Entity Framework и DataSet. Добавлены функции удаления, добавления и изменения данных к каждой таблице.

# ЛАБОРАТОРНАЯ РАБОТА №3

# «**Отображение нескольких таблиц из БД**»

Цель работы: Создать приложение, которое будет отображать данные из двух или трех связанных между собой таблиц в базе данных в формате DataGrid в среде WPF. База данных должна быть нормализована в соответствии с третьей нормальной формой. Программа должна отображать данные без возможности их изменения, удаления или добавления. Необходимо создать одну таблицу, в которой будут загружены все данные из связанных таблиц, без использования дополнительных окон, страниц или таблиц. Все названия столбцов должны быть на русском языке. Данные из различных таблиц должны отображаться в удобном виде без использования идентификаторов (ID) и прочих технических данных.

Ход работы:

* Соединение данных таблиц SQL в DataSet. Внутрь метода GetData() мы меняем запрос: правой кнопкой мыши нажимаем «Настроить» и вставляем получившийся запрос.
* **DATASET**

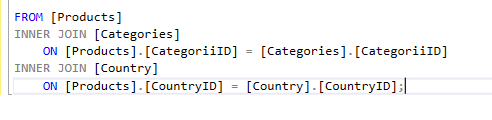


Рисунок 29- необходимый запрос в SQL.

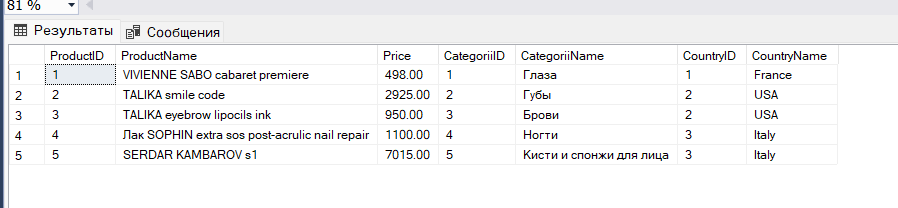


Рисунок 30- вывод запроса

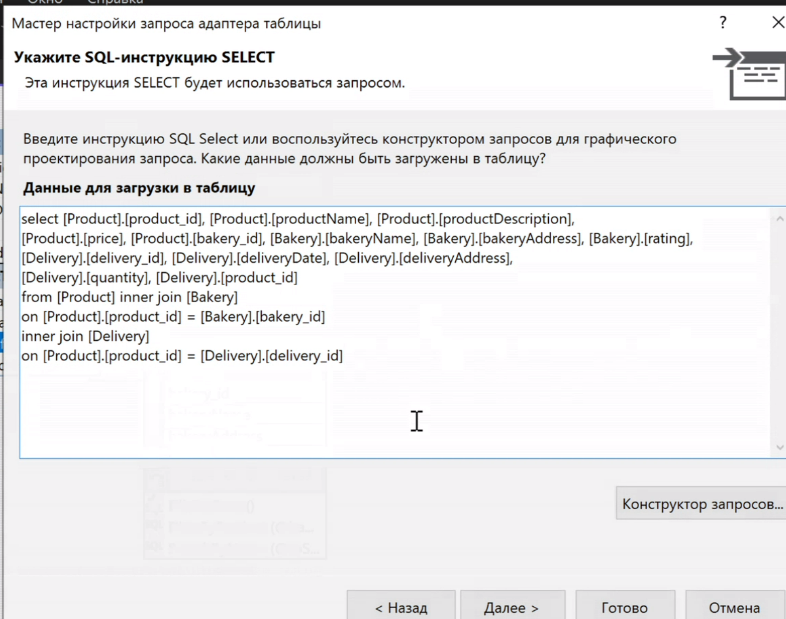


Рисунок 31- настройка запроса адаптера таблицы

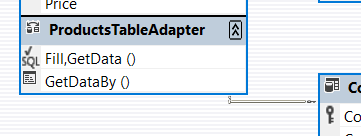


Рисунок 32- результат создания запроса

* Создаем страницу с DataGrid и отображаем данные

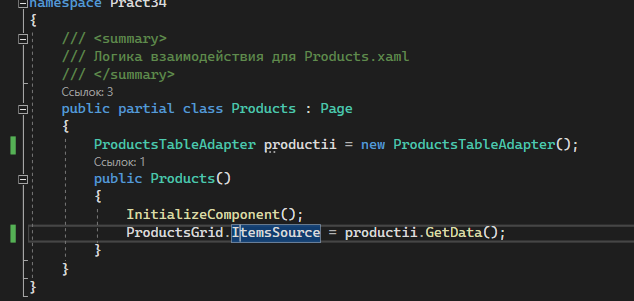


Рисунок 33- привязка к бд



Рисунок 34- DataGrid

* Для того, чтобы получить доступ ко всем столбцам из кода, мне нужно написать «названиедатагрида.Columns». Columns – массив столбцов. Выбрав определенный по индексу, я могу задать ему видимость – Visibility.Collapsed в нашем случае. Я хочу скрыть столбцы с Id. По счету, первый id это 0 номер, CategoriiID – 2 номер и CountryID – 3 номер.

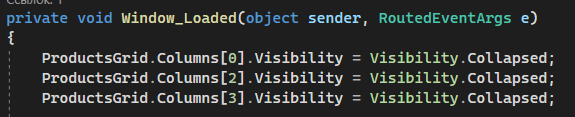


Рисунок 35- скрытие id

* Также дописываем строки «названиедатагрида.Columns». Header- это массив столбцов, название которых мы будем менять на русский язык.

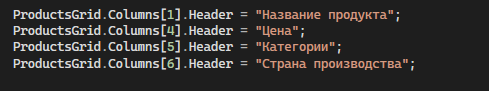


Рисунок 36- переименовываем

* НО в таком случае у нас выдается ошибка, потому что данные из БД еще не успели придти в таблицу и колонок нет. Чтобы мы были уверены, что уже все подгрузилось, давайте создадим событие Loaded для Window, и поместим код туда



Рисунок 37- Loaded

* **EntityFramework**
* В начале просто делаю выгрузку данных

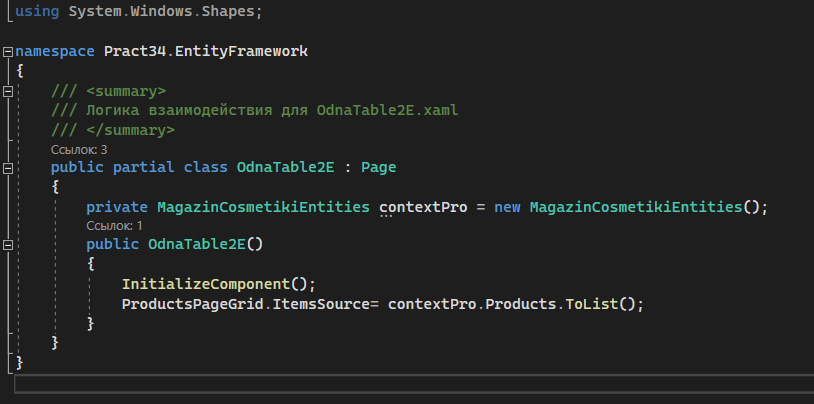
****

Рисунок 38- выгрузка данных

* Далее с помощью Binding мы создаем свои столбцы и говорим, что в определенный столбец данные будут выгружаться из определенного поля. Переименовываем столбцы на русский.

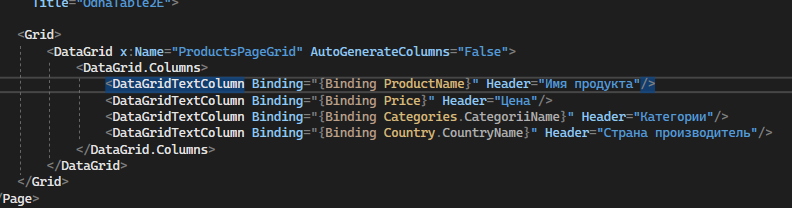
****

Рисунок 39- создание собственных столбцов



Рисунок 40- результат

Вывод: Было создано приложение, которое будет отображать данные из двух или трех связанных между собой таблиц в базе данных в формате DataGrid в среде WPF. База данных должна нормализована в соответствии с третьей нормальной формой. Программа отображает данные без возможности их изменения, удаления или добавления. Создана одна таблица, в которой загружены все данные из связанных таблиц, без использования дополнительных окон, страниц или таблиц. Все названия столбцов должны быть на русском языке. Данные из различных таблиц отображаются в удобном виде без использования идентификаторов (ID) и прочих технических данных.