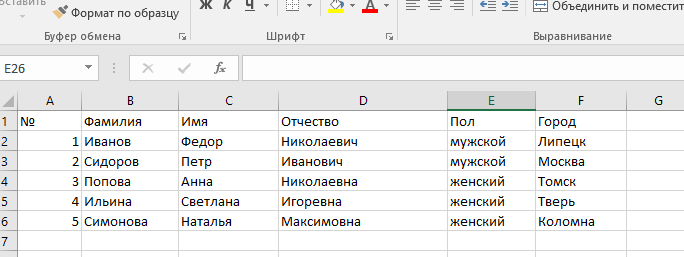
**Подключение к проекту базы данных.**

**Привязка данных из БД к элементам.**

**Переход по кнопкам между формами.**

**Видео для подготовки можно смотреть по ссылке https://nationalteam.worldskills.ru/skills/programmnye-resheniya-dlya-biznesa/**

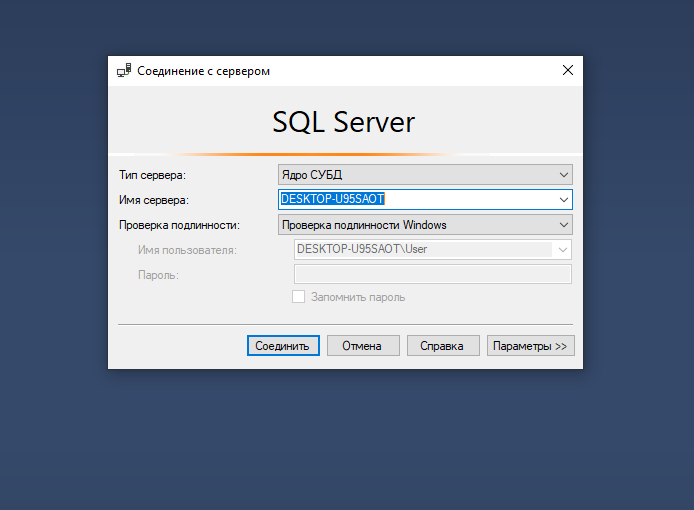
Для примера возьмем таблицу с данными



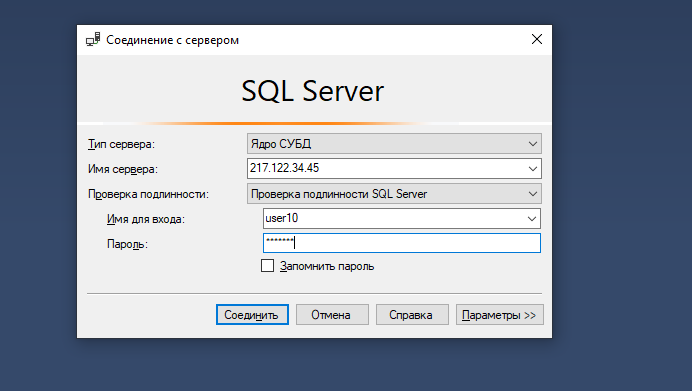
Не забываем привести таблицу к 3 нормальной форме для переноса в БД.

В базе данных создадим справочники Сотрудники, Пол и Город.

Запускаем MSSQL. Не забываем, что при работе дома на локальном сервере выбираем проверку подлинности Windows.

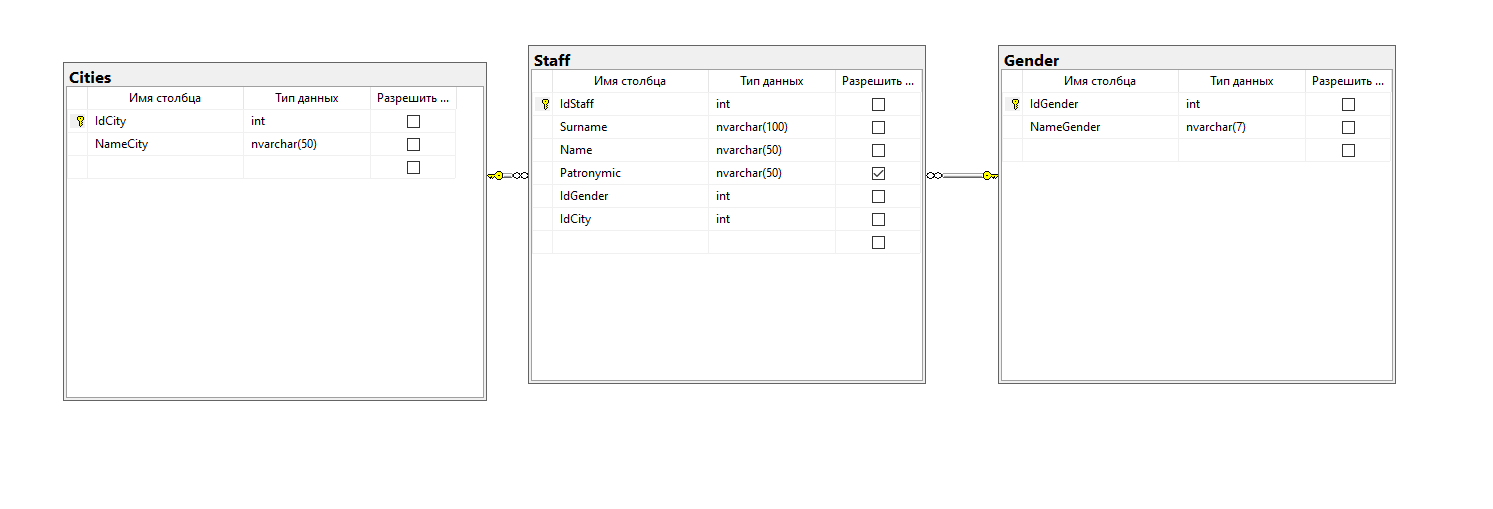


На экзамене выбираем другие настройки. Адрес сервера, логин и пароль вам сообщит главный эксперт.

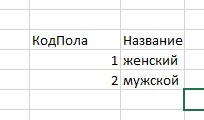
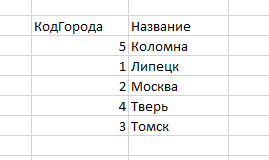


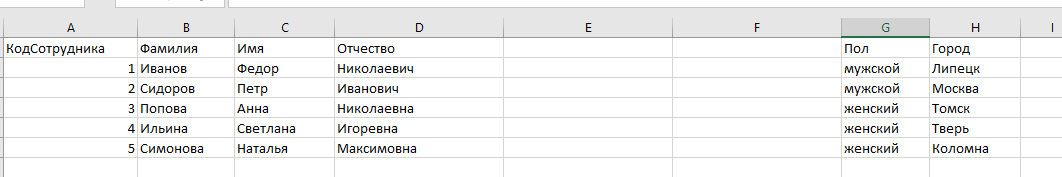
Если требуется- создаем новую базу данных. Для примера создадим базу данных PetShop.

Создаем таблицы Сотрудники, Пол и Город через схему данных.



Формируем в экселе справочники для городов и пола.



А затем подставим в таблицу Код Пола и Код Города. Для этого вставим пустые столбцы. Помним, что копирование в базу данных происходит построчно. Один столбец скопировать и вставить не получится! 

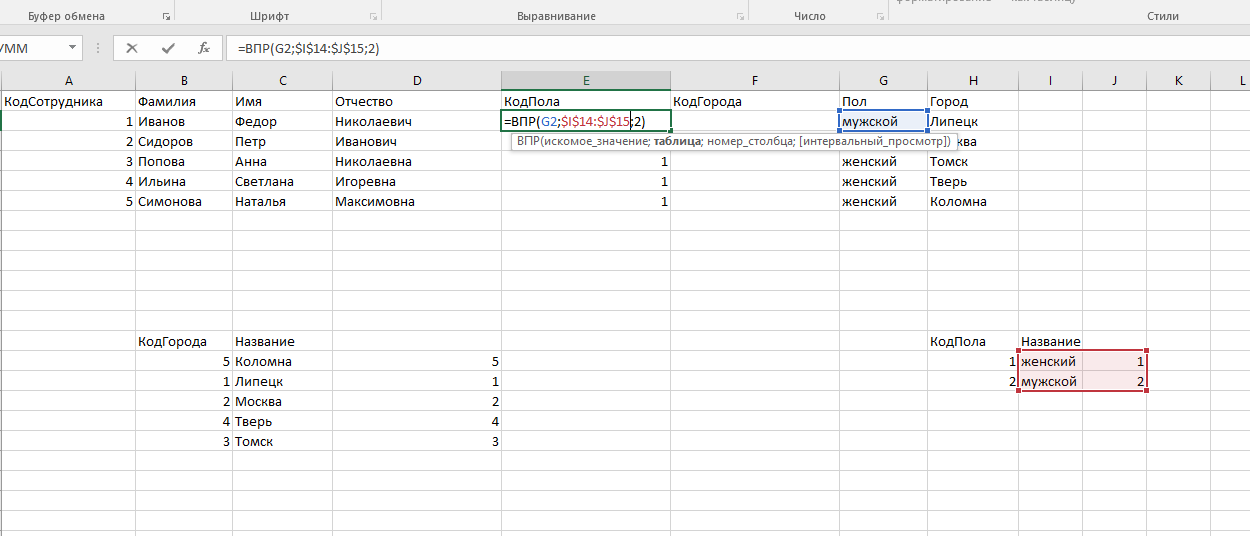
Используем функцию ВПР. Для этого не забываем в справочнике отсортировать названия по алфавиту, и продублировать номера после названий для корректной работы функции.



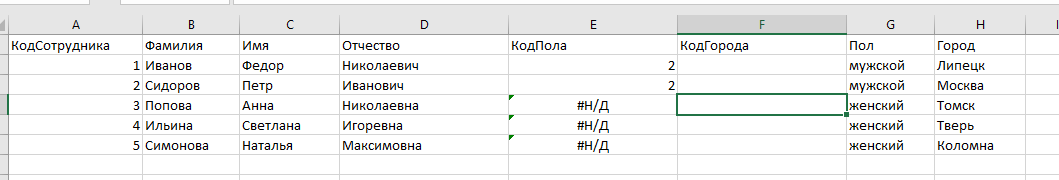
Вводим в ячейку Е2 функцию ВПР с параметрами.

Первое число (синий)- это адрес ячейки, которую будем искать в справочнике значений. Второе число (красный) – диапазон значений, среди которых ищем.

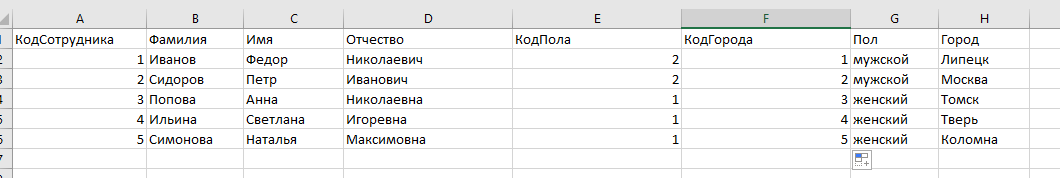
Третье число- числа из какого столбца в красном диапазоне будем поставлять, вместо найденных.



Если при использовании функции ВПР выдает ошибку, значит вы не «заблокировали» диапазон значений, и при растягивании формулы по столбцу адреса его ячеек «съезжают».

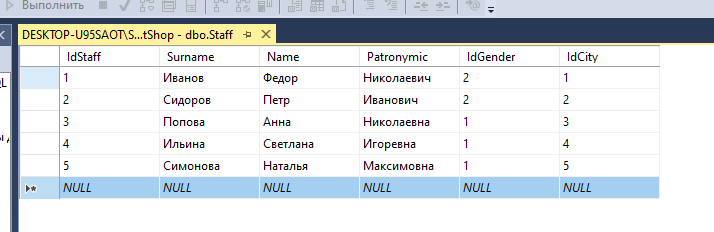


Результат:

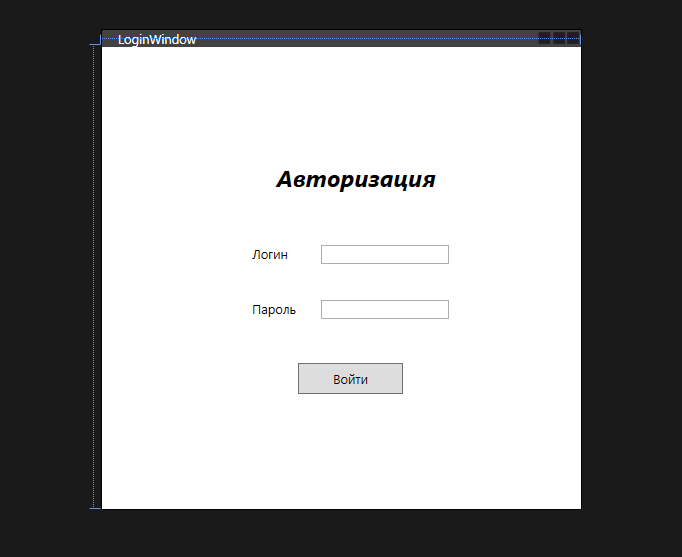


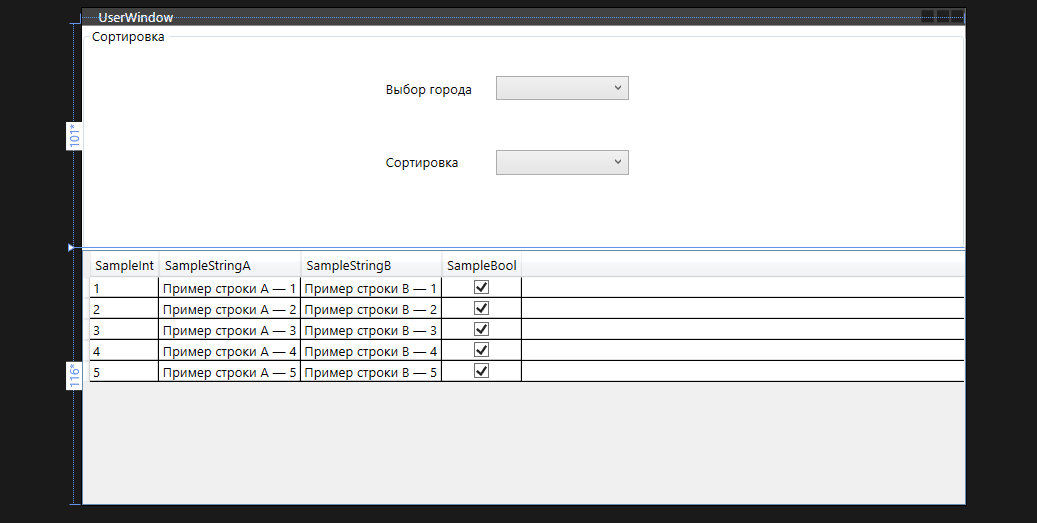
Скопируем данные в базу данных.

Результат одной из таблиц:

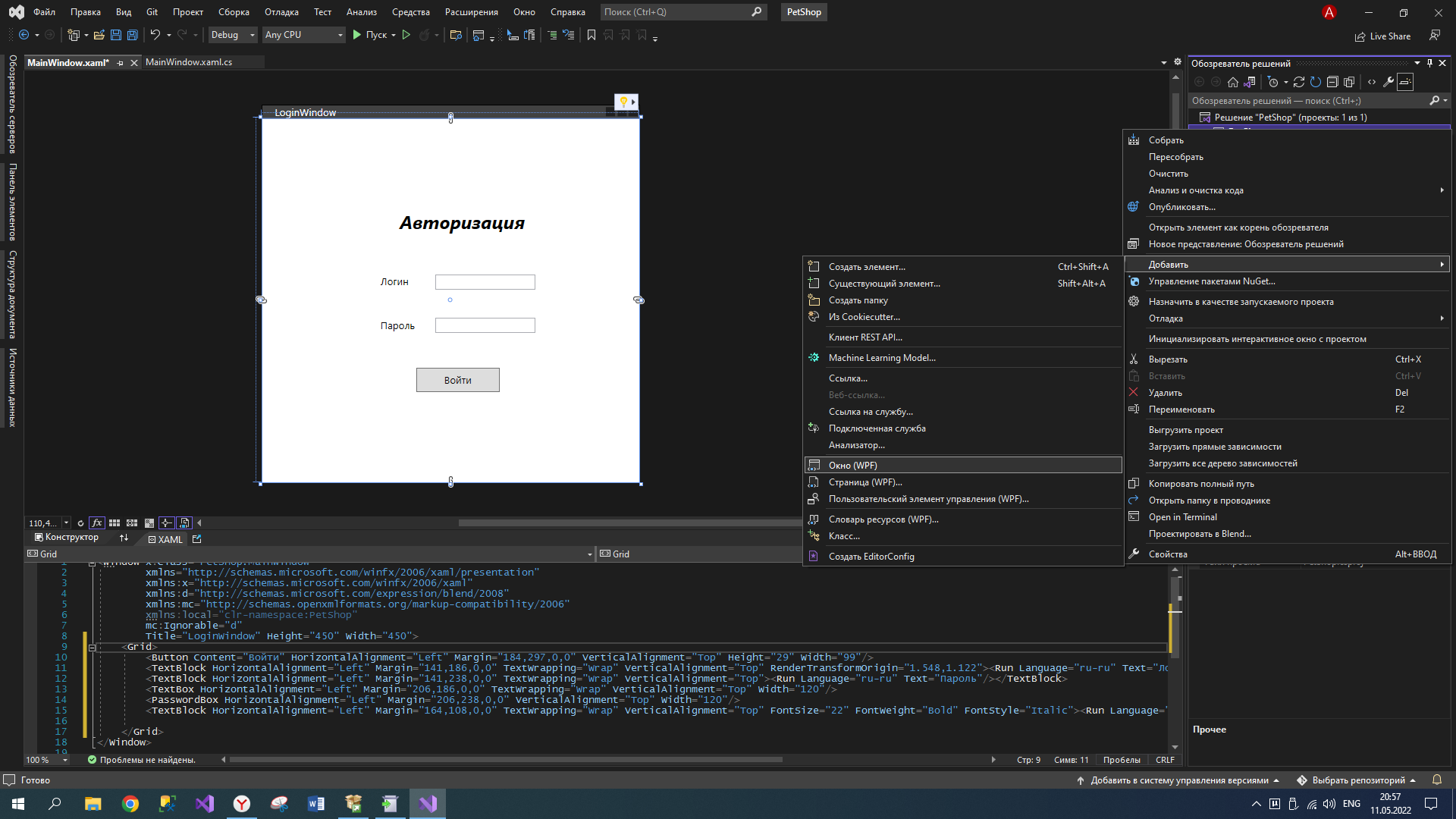


Теперь создадим настольное приложение, состоящее из двух форм.

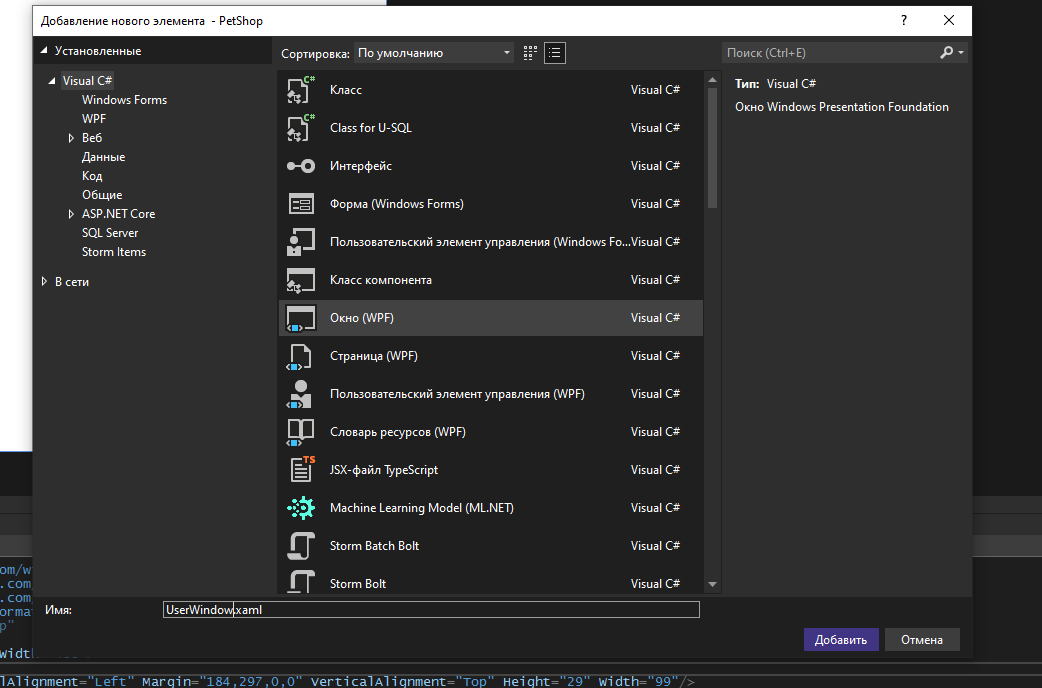




Создание второй формы: Щелкаем правой кнопкой мыши по названию проекта слева. Добавить- Окно.



Вводим имя окна

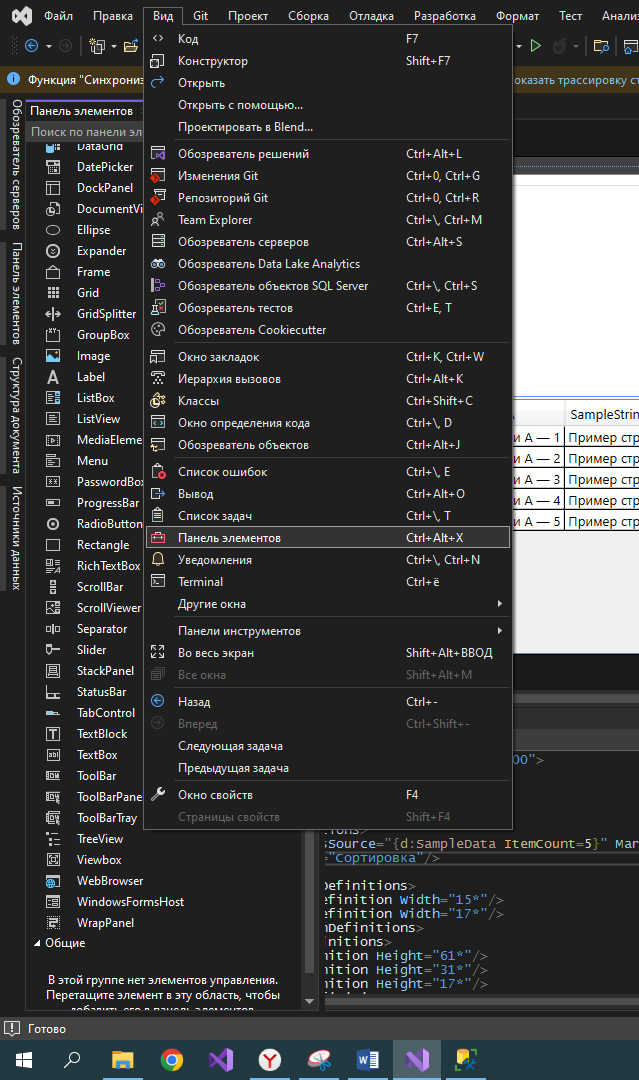


Во втором окне мы использовали компоненты groupbox для группировки верхних выпадающих списков (combobox).

И компонент Datagrid для отображения данных из БД.

Все они доступны в боковой панели Панель элементов.

Если ее нет, откройте с помощью вкладки Вид-Панель элементов.

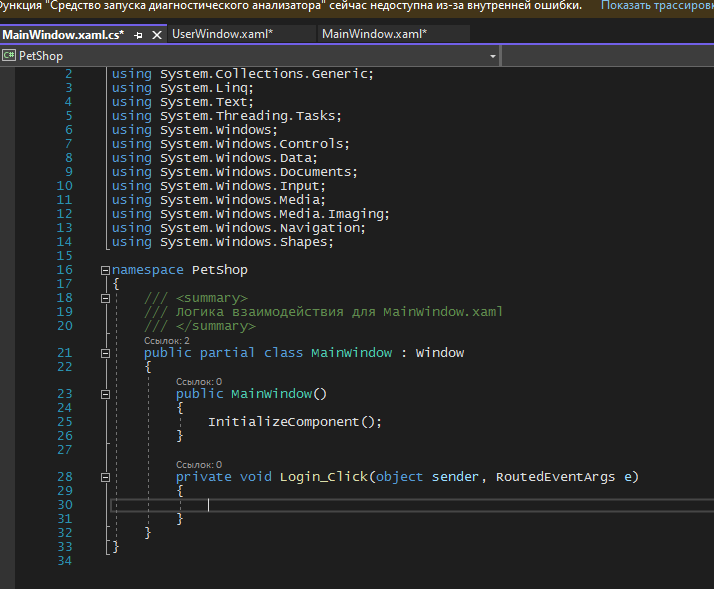


Присвойте выпадающим спискам, кнопке и датагрид имена, для того, чтобы можно было к ним обращаться из кода.

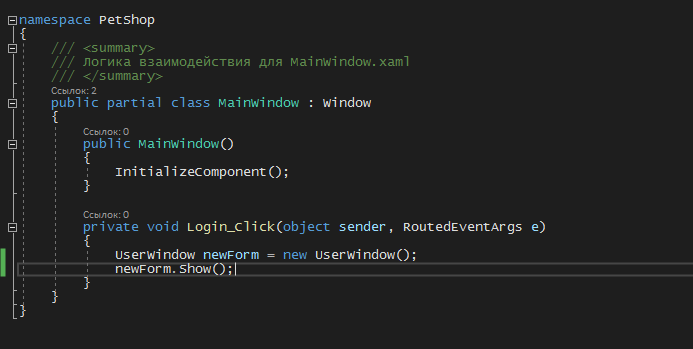
У кнопки также напишите свойство click



Два раза щелкните мышкой по нему и нажмите клавишу F12, сформируется метод, в котором мы будем писать код для перехода с формы авторизации на вторую форму.



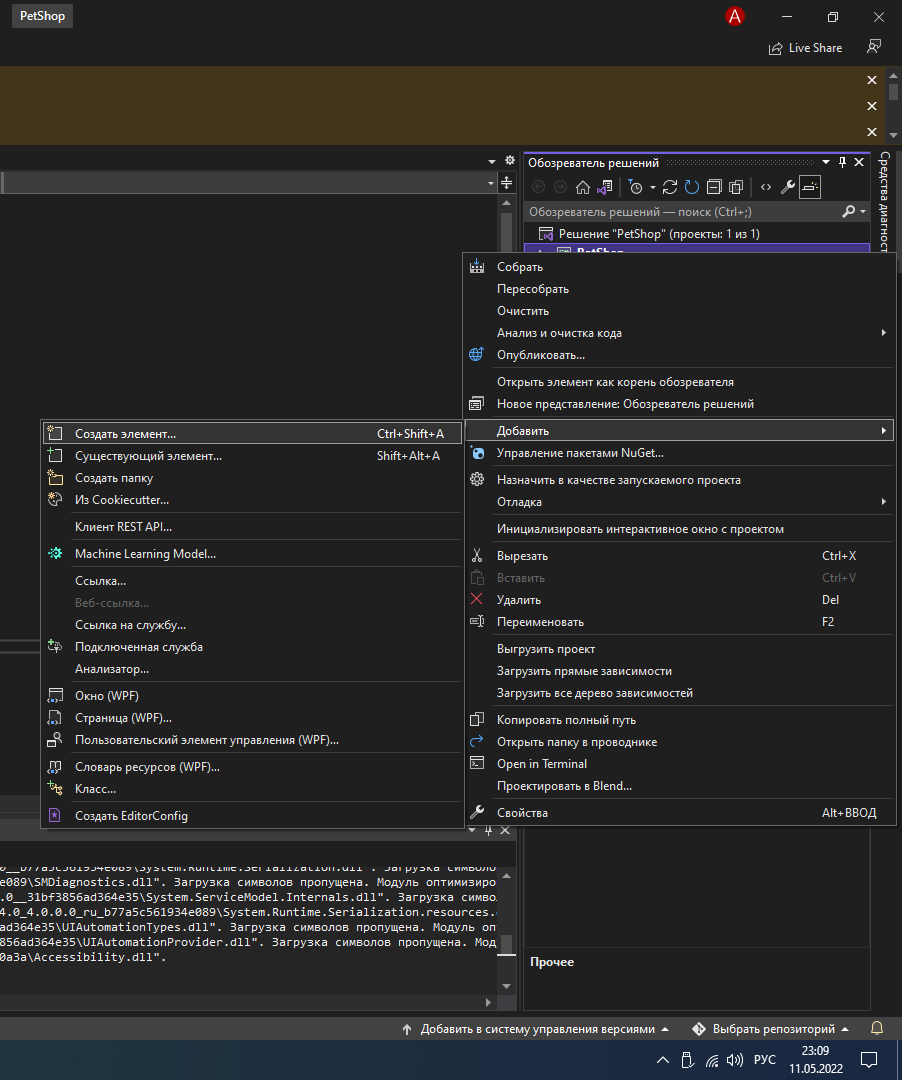
Теперь добавим в него код вызова второй формы. У нас вторая форма называется UserWindow, поэтому сначала мы создаем объект данного класса, а потом для его отображения на экране вызываем метод Show:



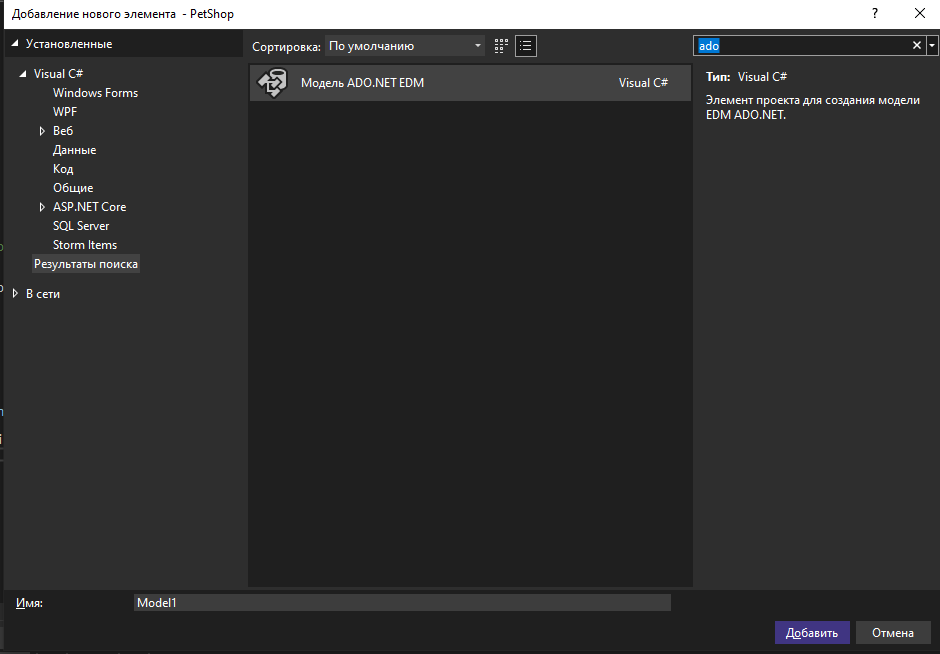
С переходами закончили.

Теперь настроим подключение БД к нашему проекту.

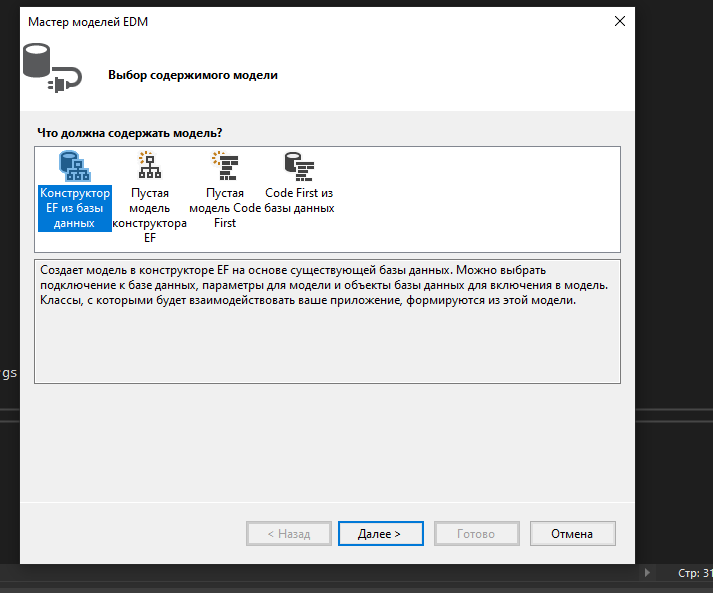
Щелкаем на имени проекта справа и выбираем Добавить- Создать элемент



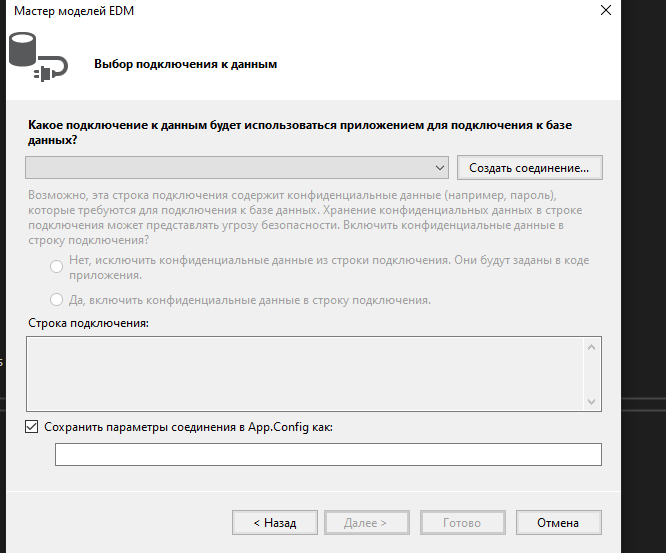
В списке выбираем нужный и жмем Добавить.



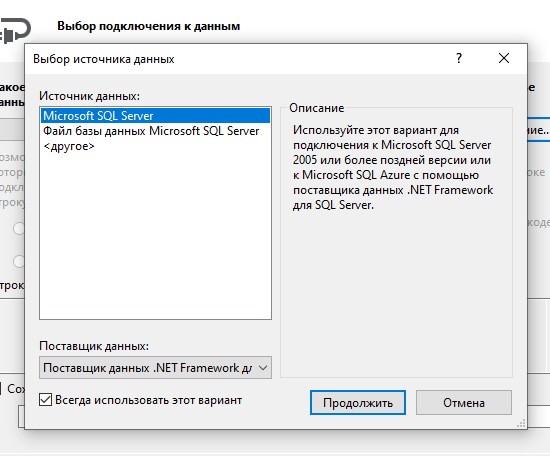
Далее следуем инструкции на картинках



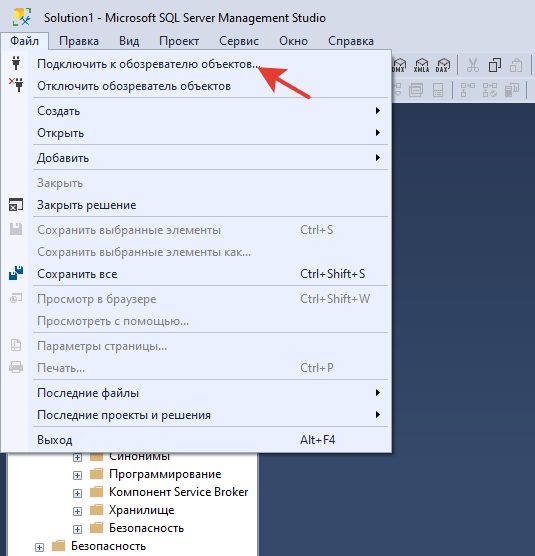
Жмем кнопку Создать соединение



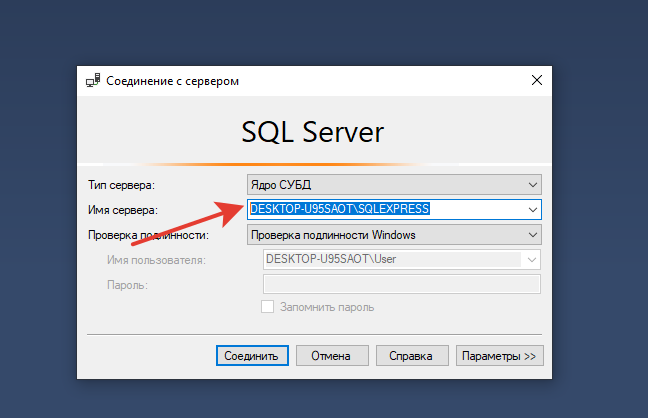
Выбираем то, что на картинке.



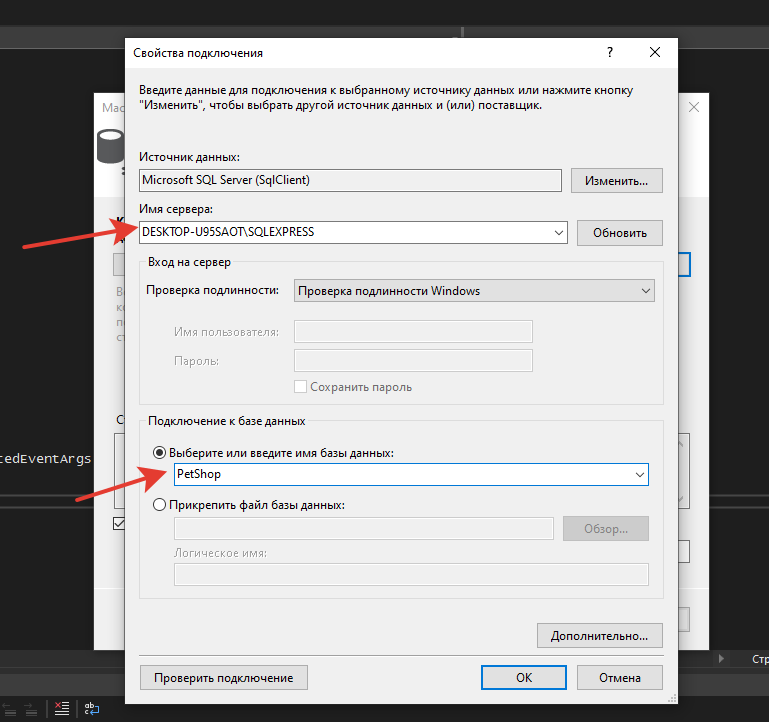
Теперь нам нужно ввести имя сервера. Возвращаемся к нашей бд и посмотрим его.



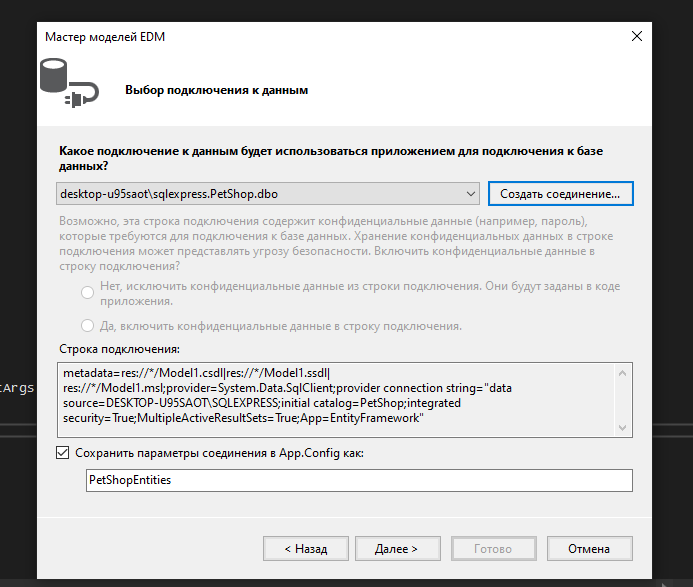
Вот это имя копируем



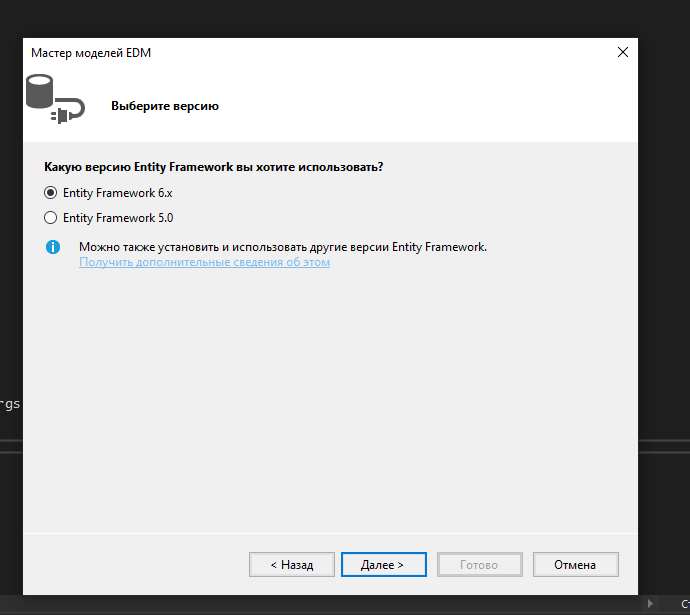
Вернулись в VisualStudio и вставили. Из выпадающего списка выбрали имя своей БД. И нажимаем ОК.



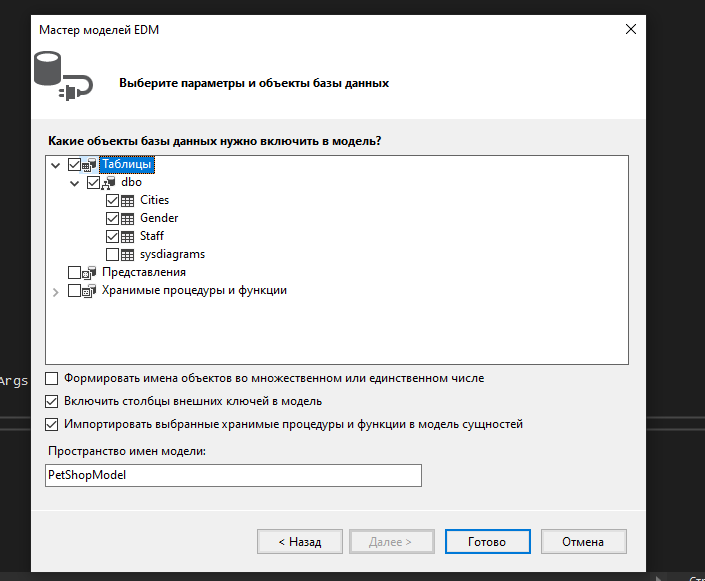
Жмем Далее.



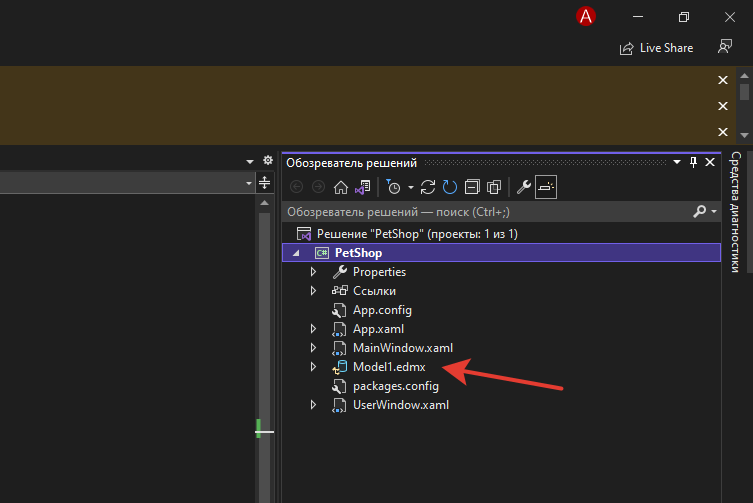
И снова Далее.



Ставим галочки напротив таблиц, которые вы создали. И жмем Готово.



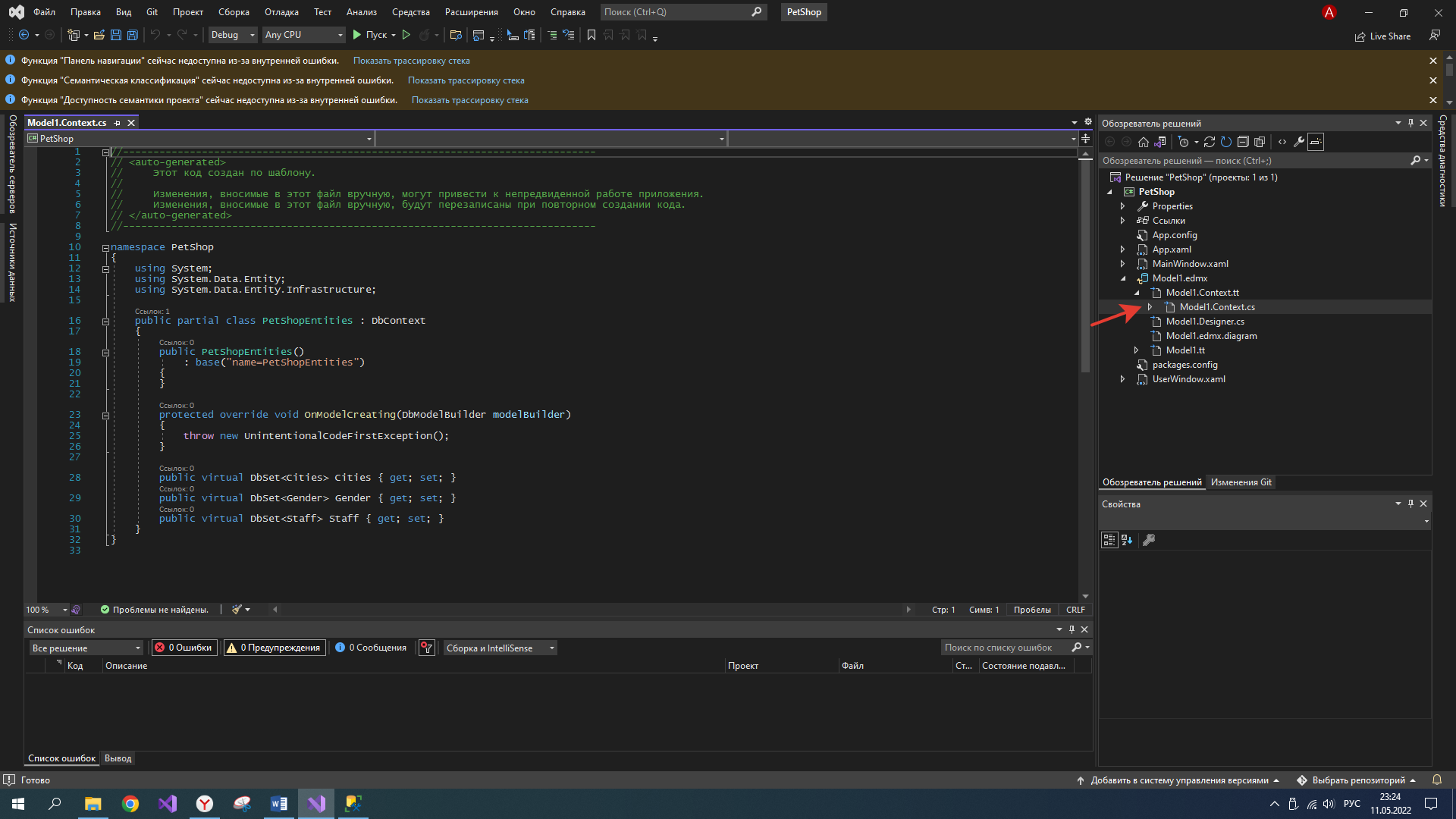
Смотрим в обозреватель решений. Если появился нужный файл, поздравляю! Вы подключились к БД.



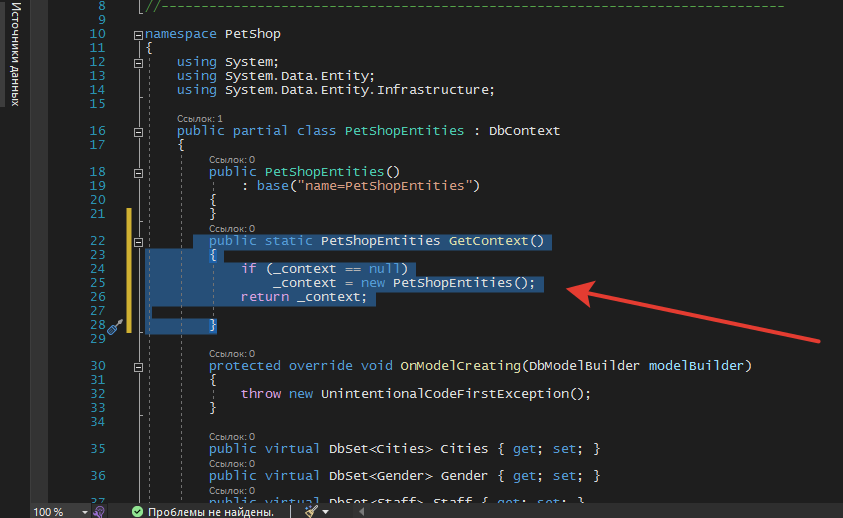
Далее выведем в датагрид данные из таблицы с сотрудниками. Заметьте, там есть и подстановочные поля.

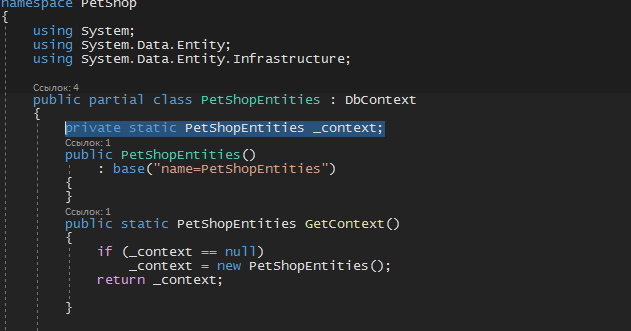
Обращение к модели данных из БД.

Открываем файл Model1.Context.cs (если вы называли модель Model1, иначе ищите по своему имени, но часть в имени .Context.cs останется такой же)



Добавляем метод получения экземпляра этого контекста. Т.е. данных из БД.





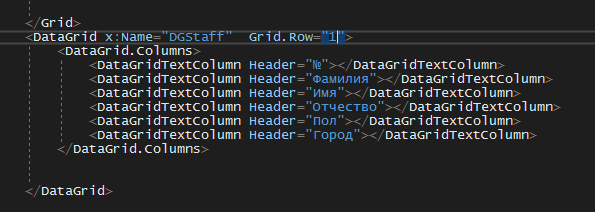
Теперь вернемся к окну UserWindow и подготовим DataGrid к выводу данных.

Список данных выводится по столбцам, которые прописывает разработчик. Устанавливаем их с помощью свойства DataGrid.Columns, которое и описывает набор столбцов.

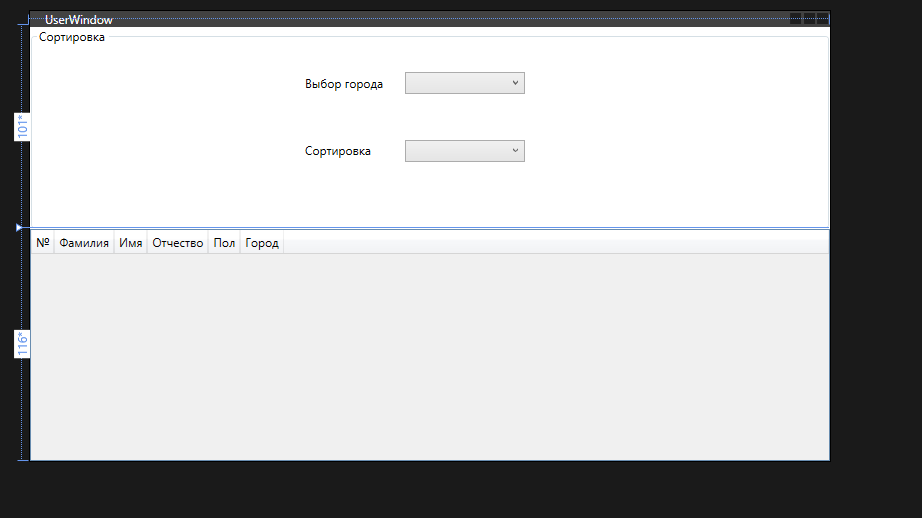
* DataGridTextColumn для текстовых столбцов
* DataGridTemplateColumn для более сложного представления данных, например, кнопки

Разметим датагрид на столбцы и присвоим им имена

Код

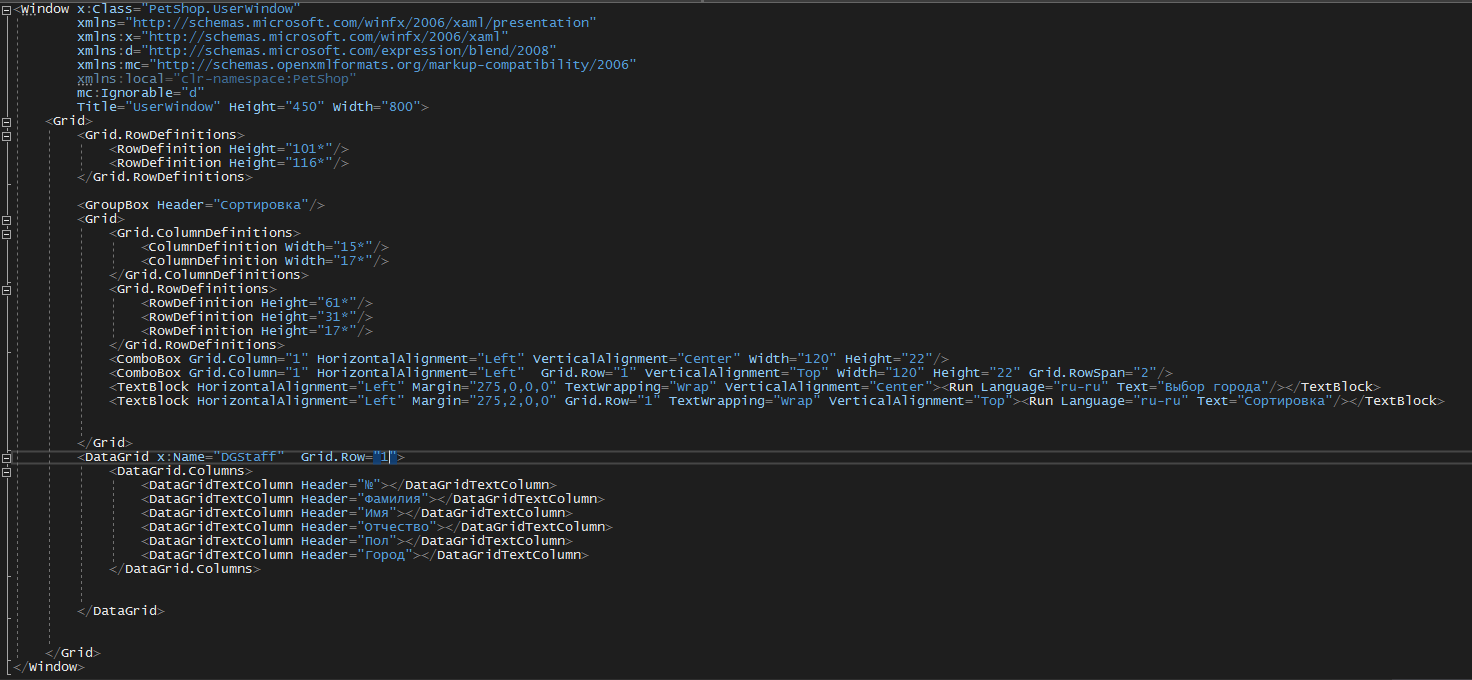


Вид формы

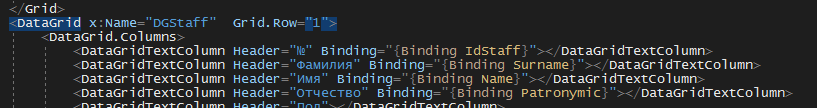


Пока пусто и не очень красиво…

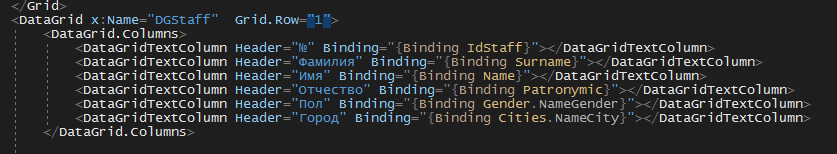
Общий код формы.



Далее сделаем привязку данных к столбцам таблицы через свойство Bind. С обычными данными всё просто. Открываем БД и смотрим как называются столбы, чтоб использовать их в свойстве Bind. Редактор кода тут предлагать ничего не будет, нужно писать названия руками очень внимательно.



Привязка столбцов с подстановкой.

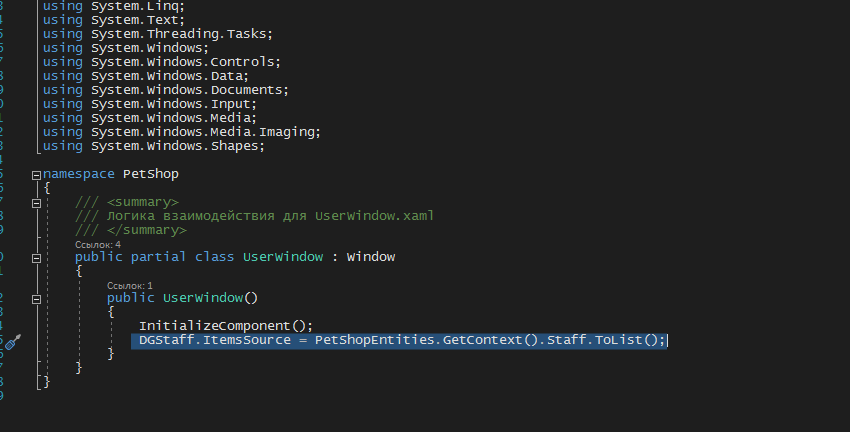


Далее загрузим список сотрудников в коде в таблицу:

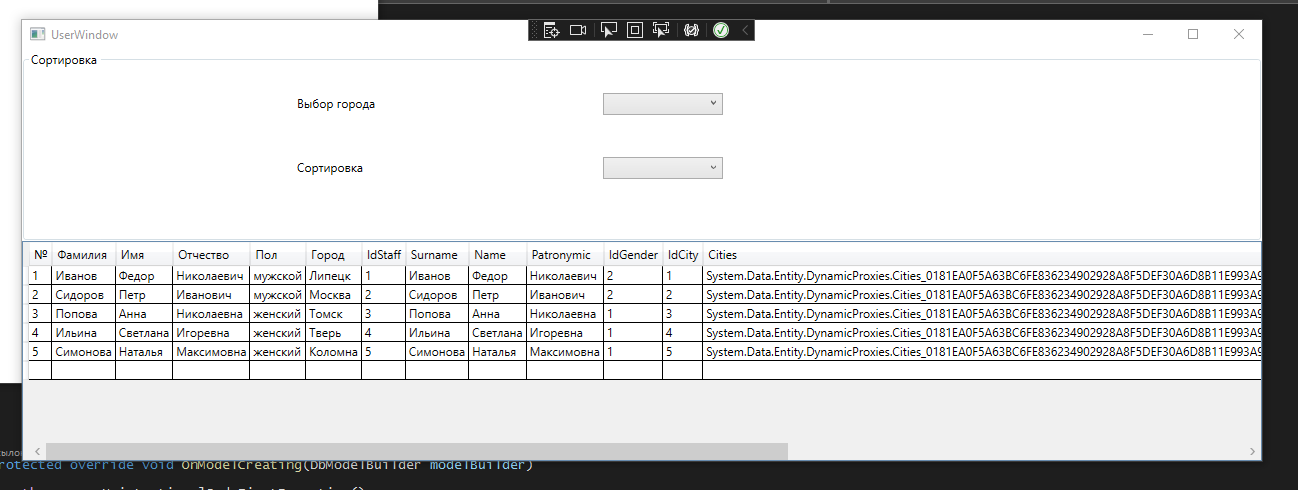
a) Нажимаем F7

b) Обращаемся к контексту модели

Дописываем выделенную строчку и запускаем приложение



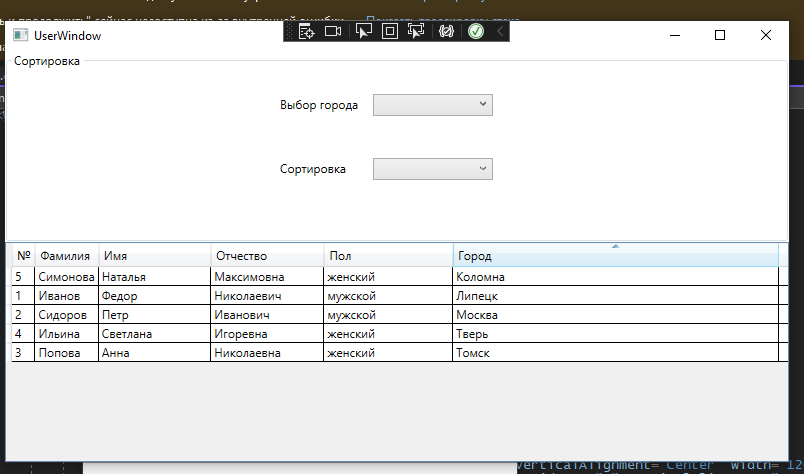
Результат. Видим, что много лишних столбцов.



Для отключения загрузки всех свойств объектов, необходимо прописать AutoGenerateColumns="False" и IsReadOnly="True"



Результат:



Ширину столбцов при выводе можно регулировать с помощью свойства Width, как и у других элементов.

Теперь поработаем с выпадающими списками. Список Сортировка будет с фиксированным набором значений. Данные списка Выбор города подтянем из базы данных.

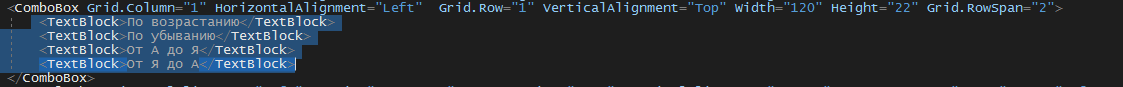
Не забываем дать им имя, чтобы потом можно было программно к ним обратиться.



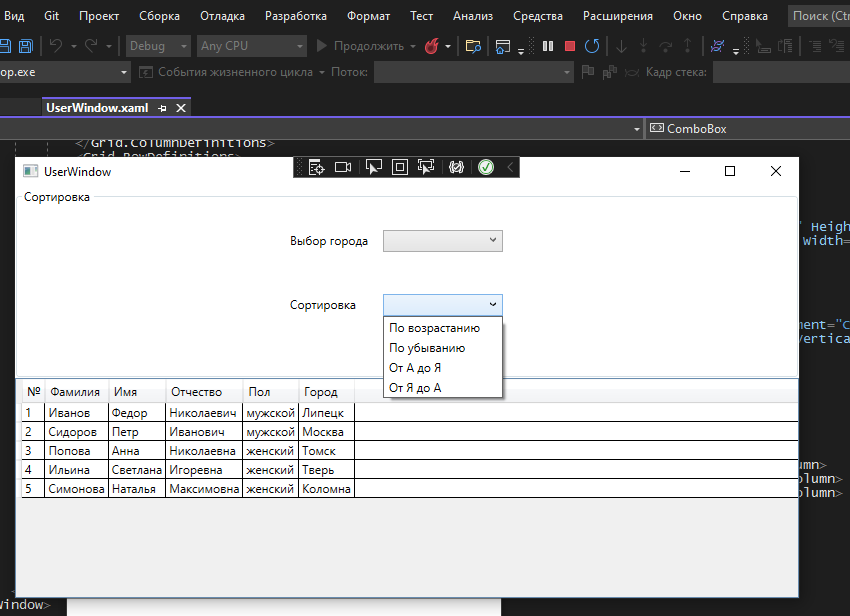
Чтобы ввести фиксированный набор значений нужно добавить закрывающий тег к комбобоксу. Для этого убираем слеш в конце скобки и добавляем закрывающий тег.



Затем добавляем значения между тегами:



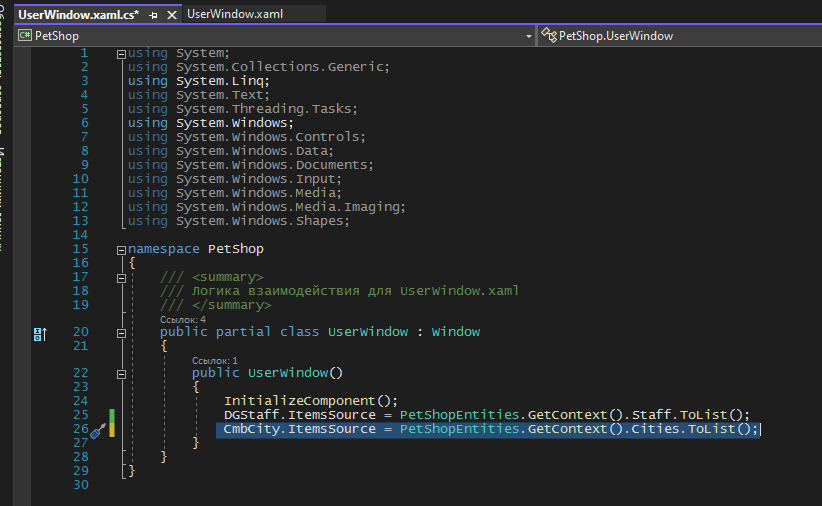
Результат



Теперь выводим города во второй список. Используем свойство DisplayMemberPath с указанием имени столбца в нужной таблице.



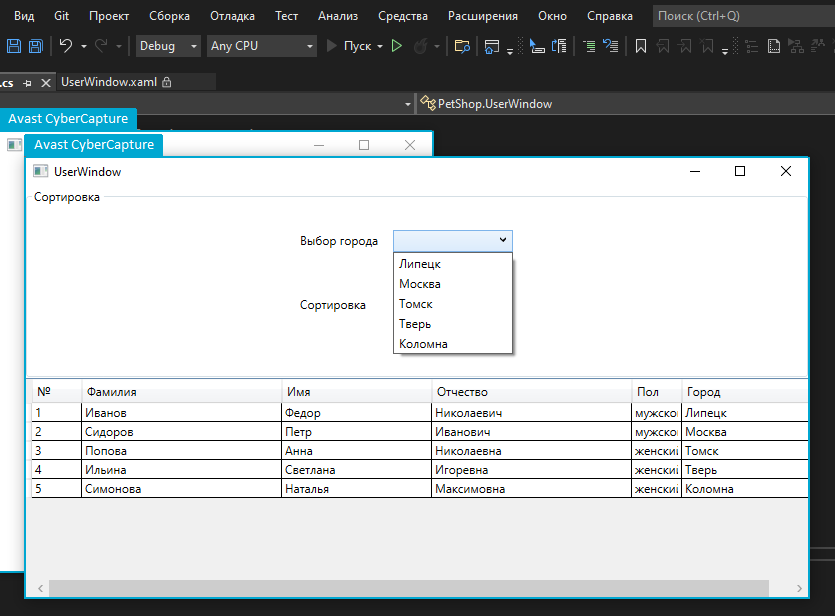
Далее дописываем код для забора данных из БД.



CmbCity.ItemsSource – свойство значения пунктов списка

PetShopEntities.GetContext().Cities.ToList() – Забираем данные из БД PetShopEntities с помощью функции GetContext() из таблицы Cities выводим списком ToList()

Запускаем- проверяем



Хвалим себя и запоминаем код.