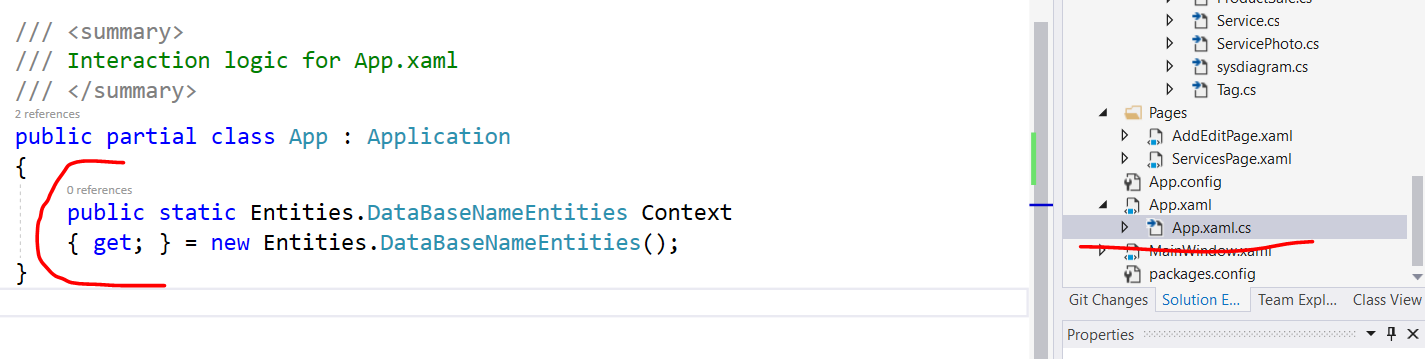
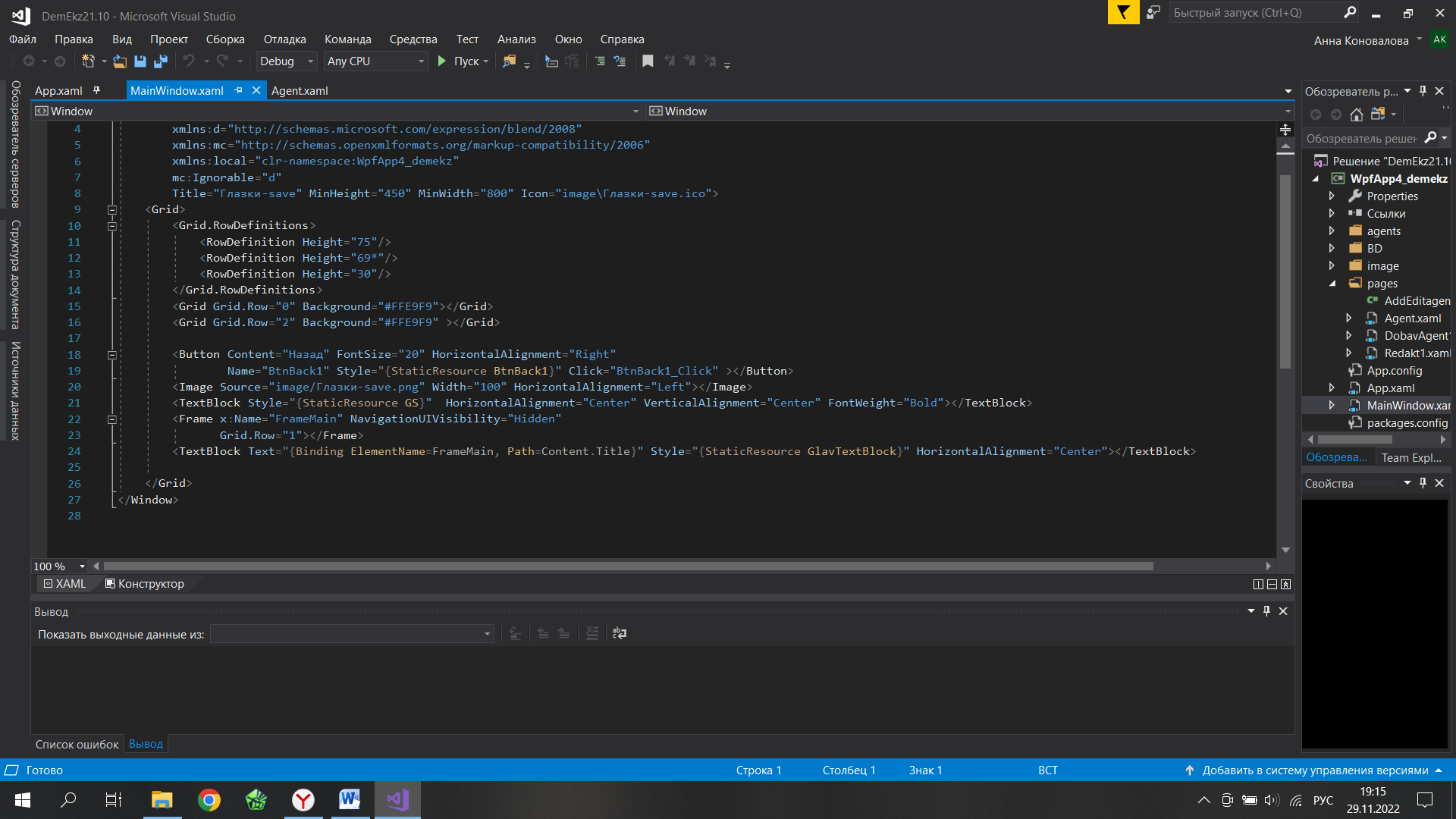
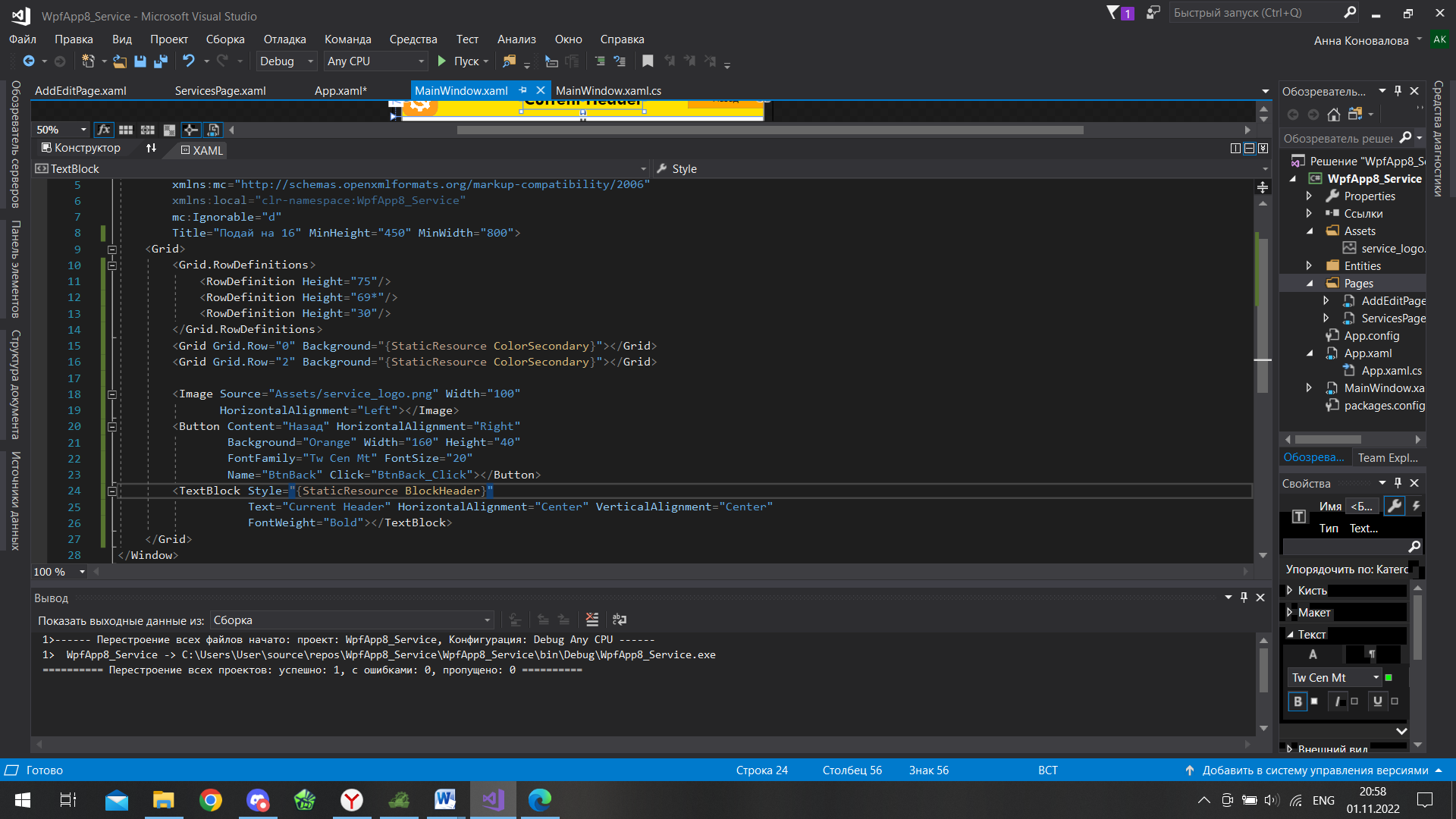
1. ПОСЛЕ ТОГО КАК ДОБАВИЛИ БАЗУ ДАННЫХ В VISUAL

**В APP.XAML.CS ПИСАТЬ ЭТО**



2.ГЛАВНАЯ СТРАНИЦА





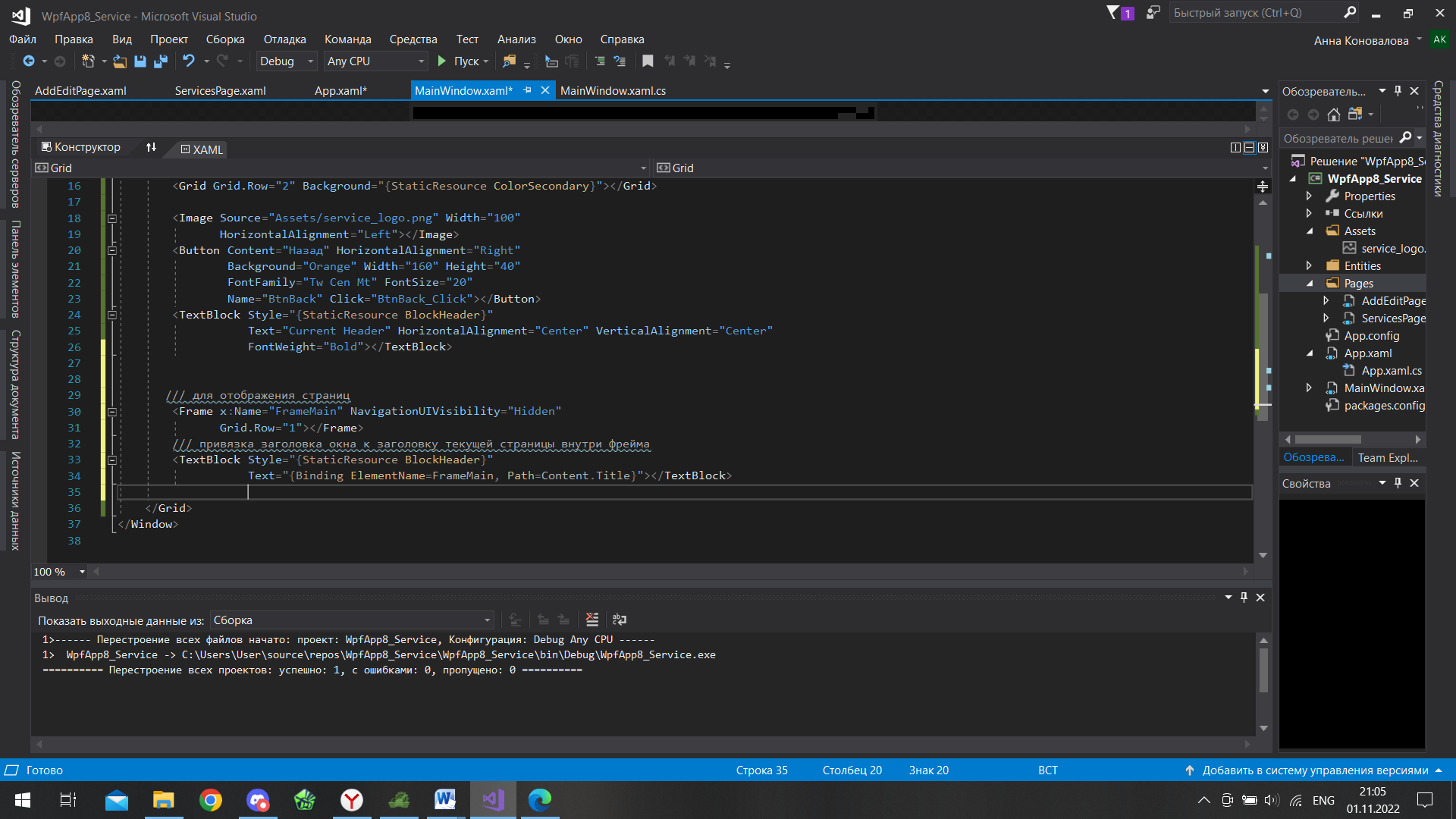
СТИЛЬ

<Brush x:Key="ColorSecondary">#FFE9F9</Brush>

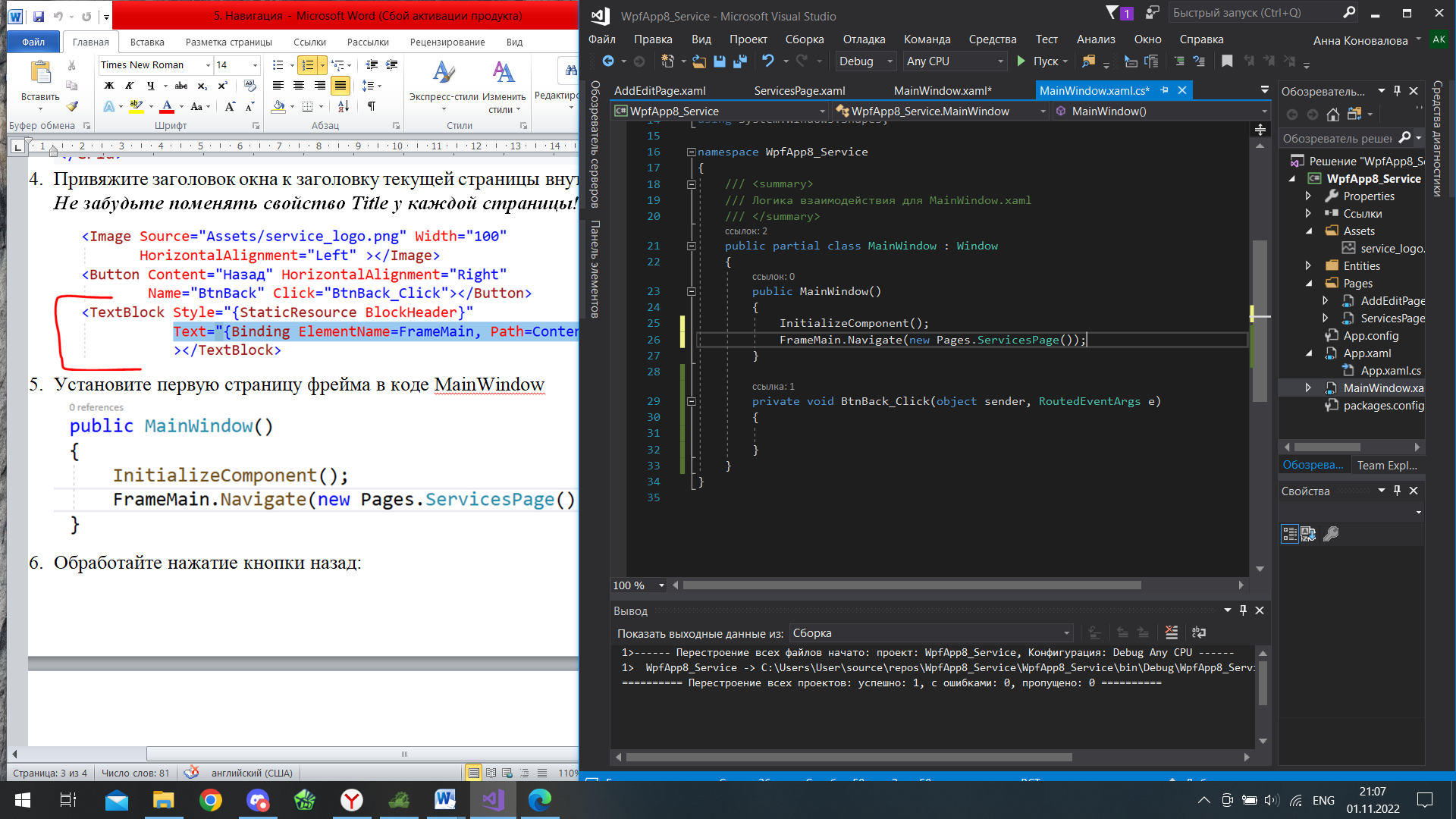


ДЛЯ GRID

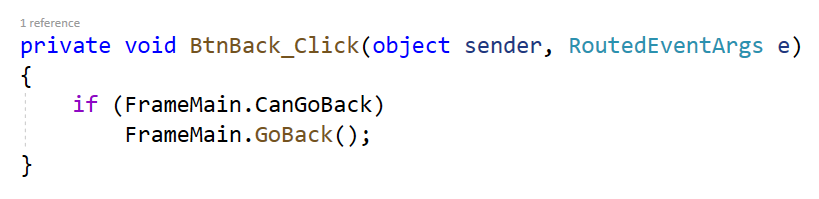
3. FRAME ДЛЯ ОТОБР СТРАНИЦЫ И ПРИВЯЗКА ЗАГОЛОВКОВ

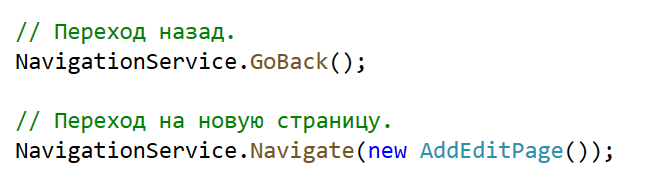


4.УСТАНОВКА 1-Й СТР ФРЕЙМА В КОДЕ MAIN WINDOWS

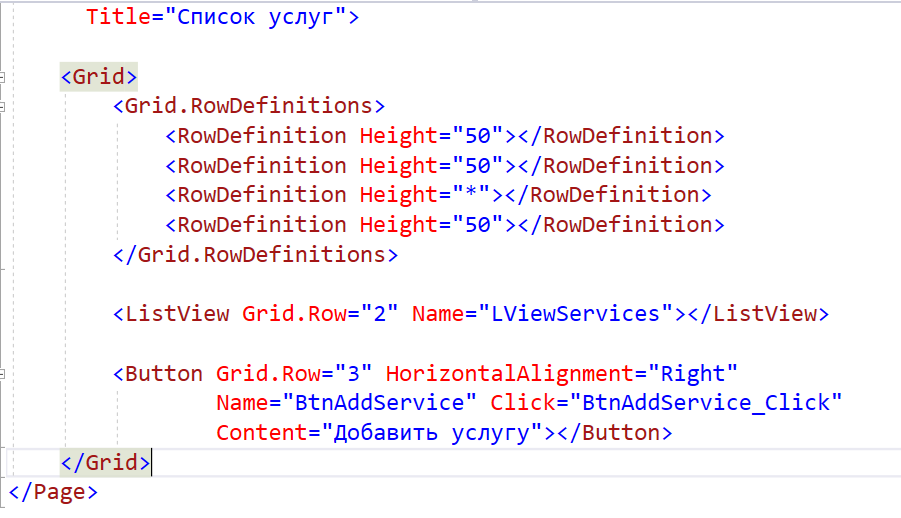


5.КНОПКА НАЗАД





6.следующая страница



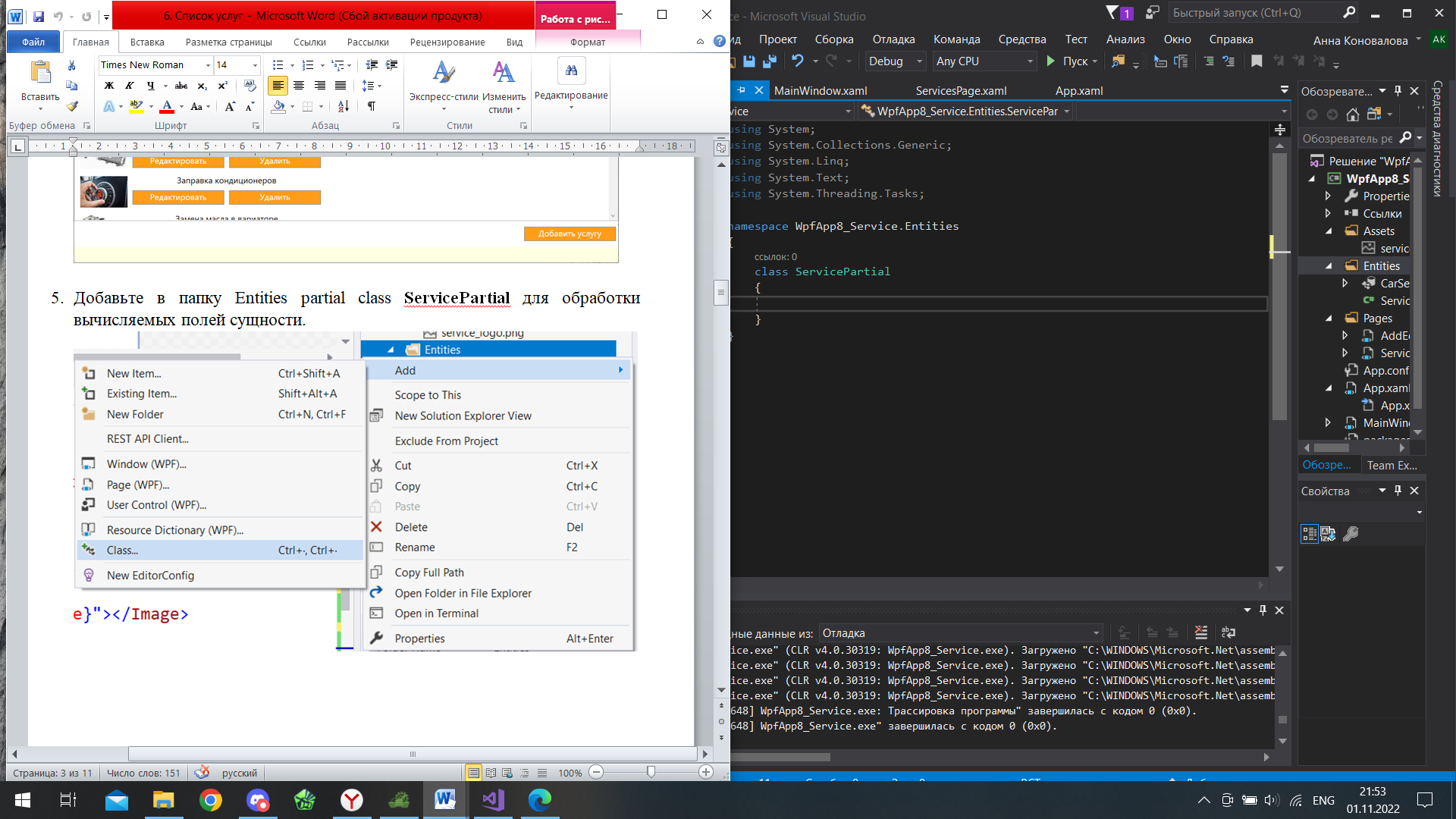
7. отображение элементов **внутри ListView** (первый этап – изображение, название, кнопки)

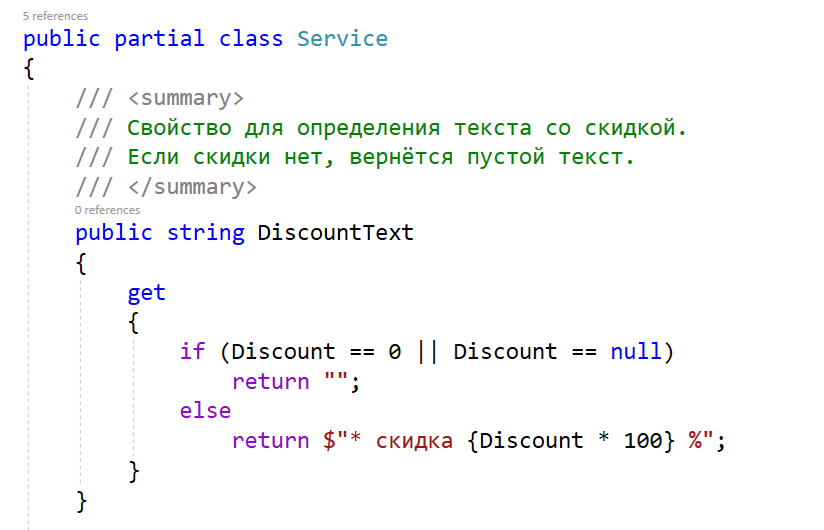


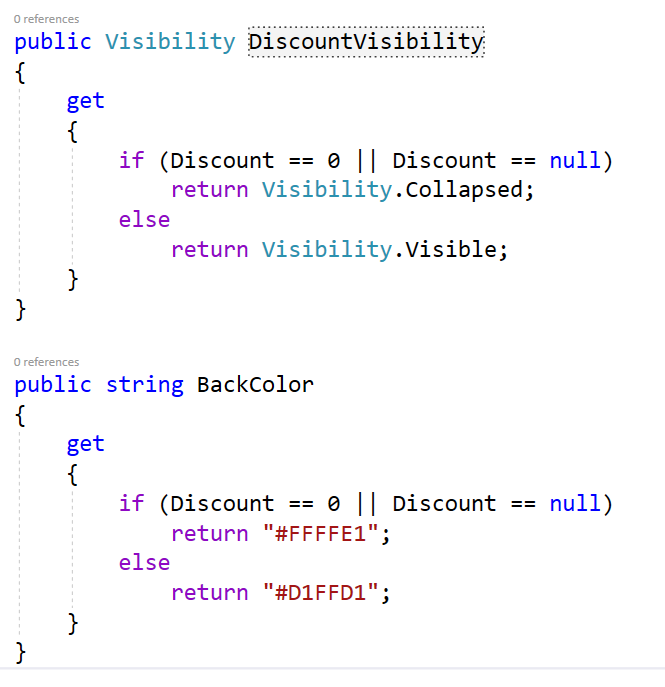
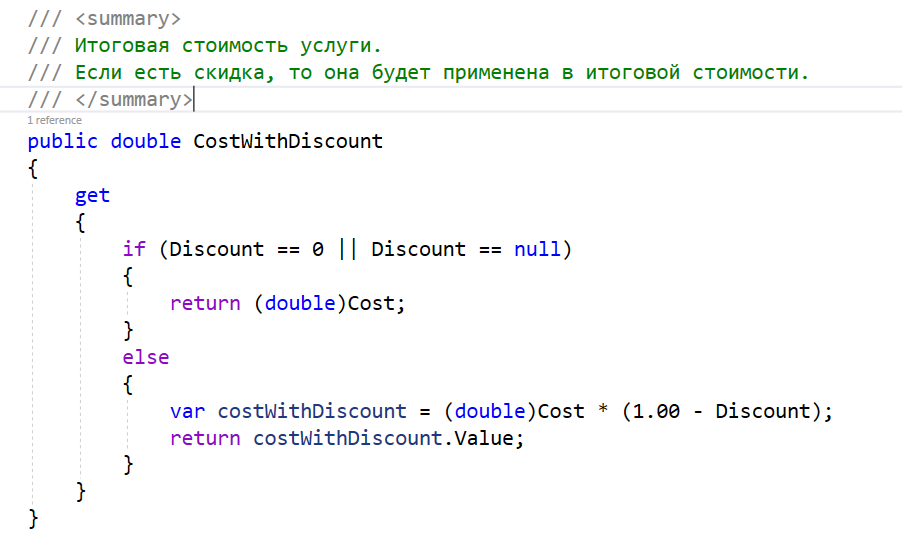
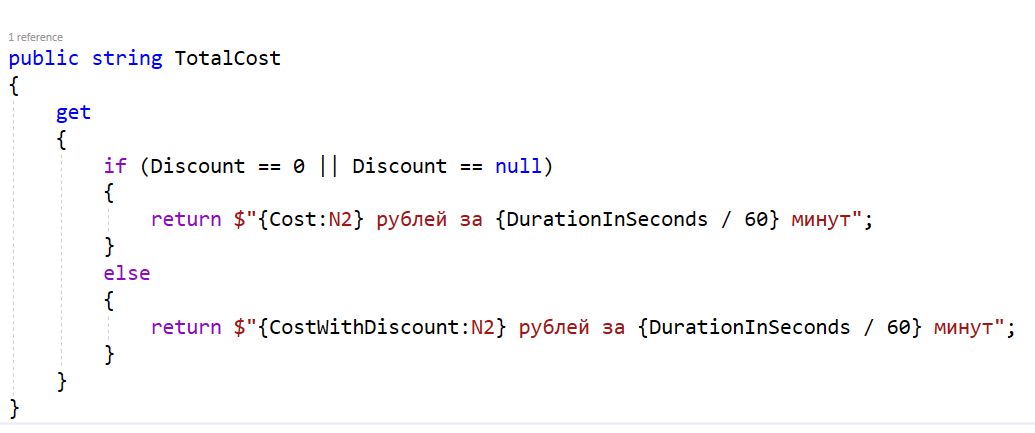
8. Загрузите список услуг из базы в ListView



9.В ПАПКУ БАЗЫ ДАННЫХ добавить класс



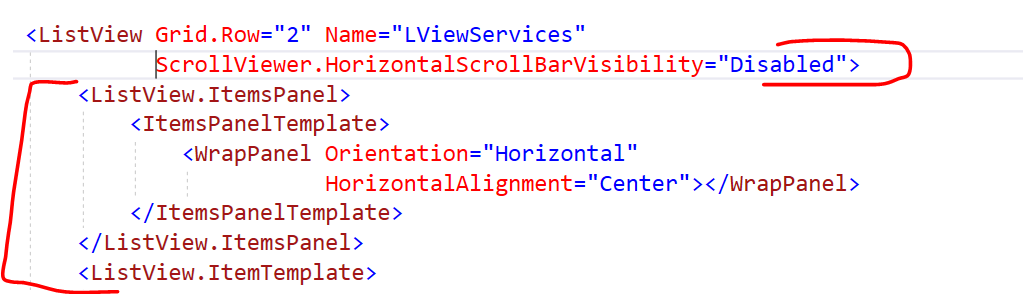
СКИДКА



10. Измените верстку услуги в списке, таким образом, чтобы она использовала новые свойства (в соответствии с заданием).



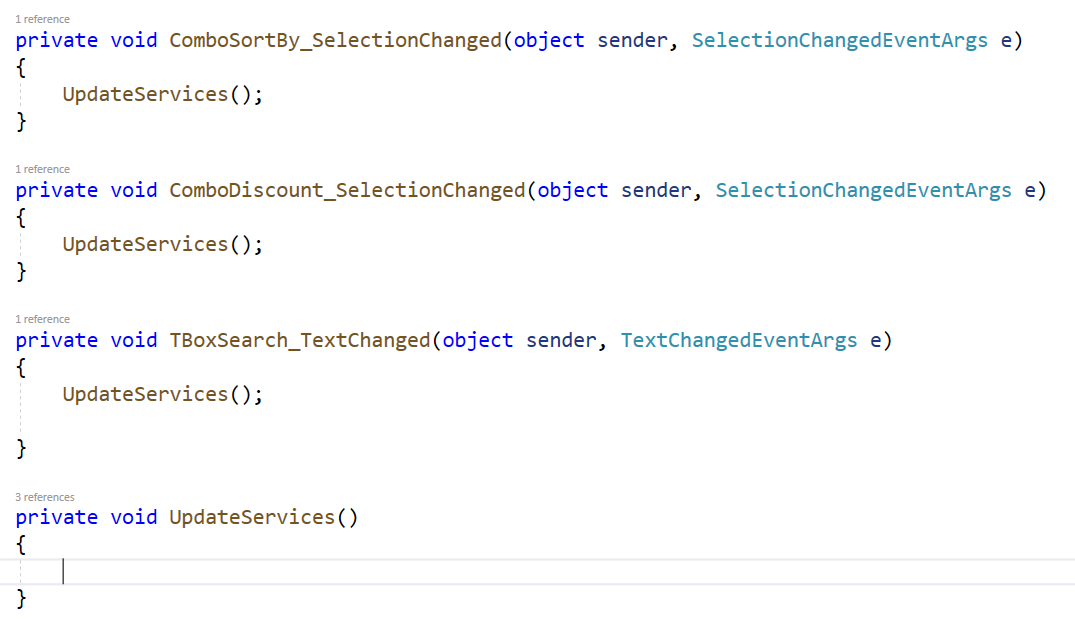
11. Измените контейнер расположения элементов на WrapPanel.

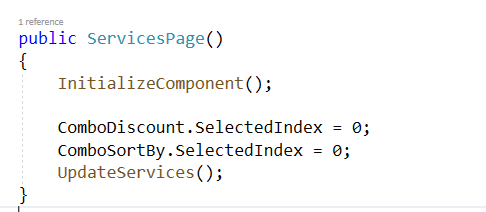


12. ФИЛЬТРАЦИЯ, СОРТИРОВКА, ПОИСК

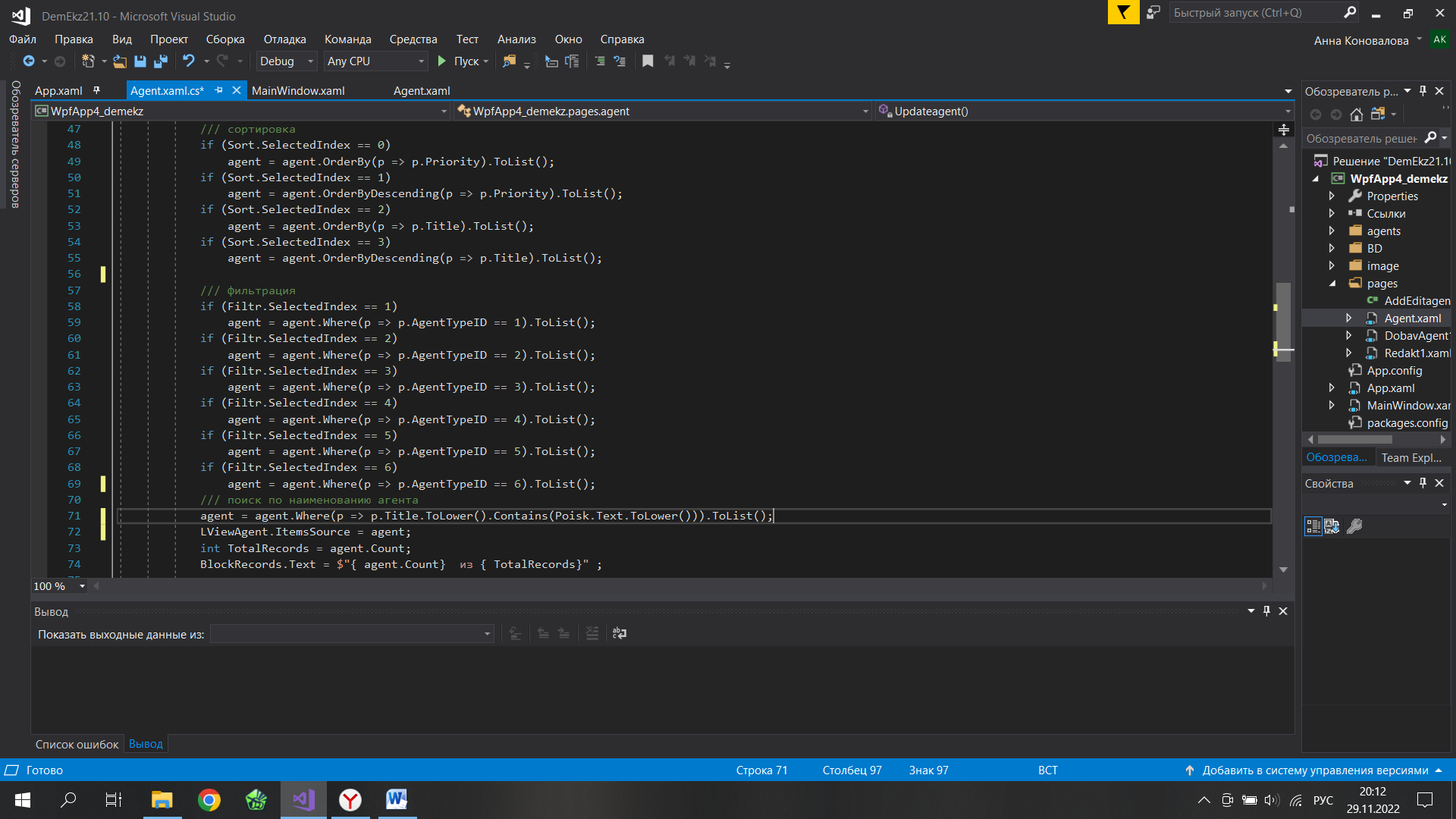


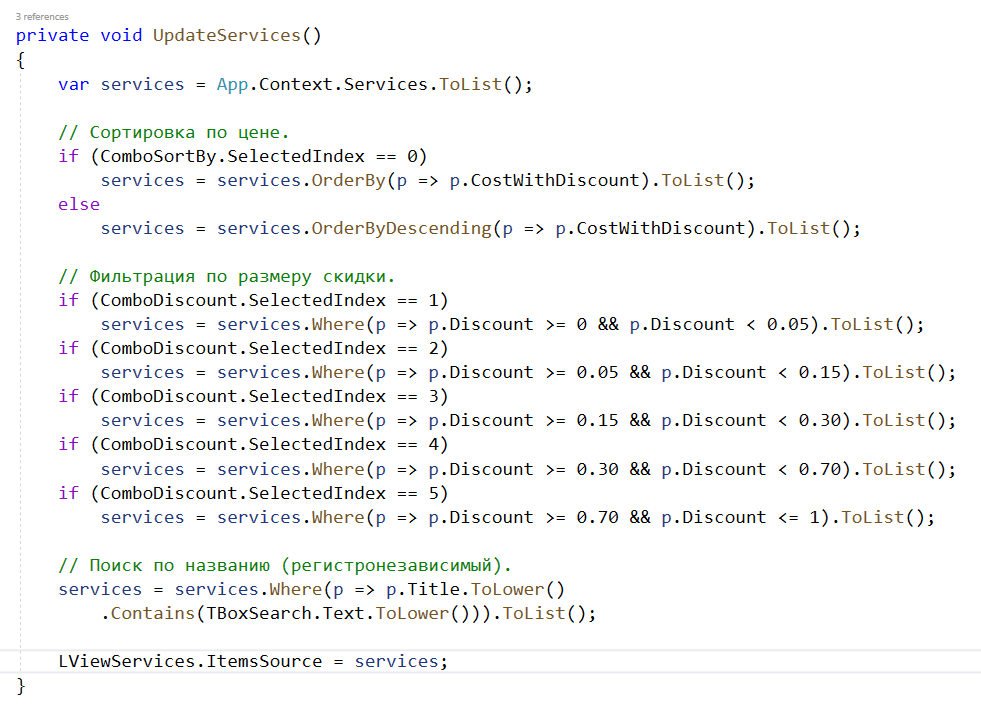
Обработайте срабатывание событий фильтров следующим образом: каждое событие вызывает общий метод UpdateServices, внутри которого все необходимые записи проходят условия отбора и затем выводятся в ListView.





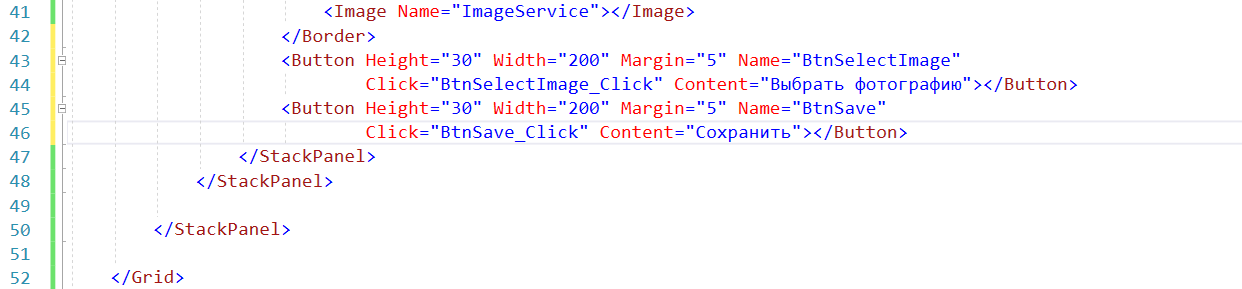
В методе UpdateServices пропишите обработку необходимых фильтров.



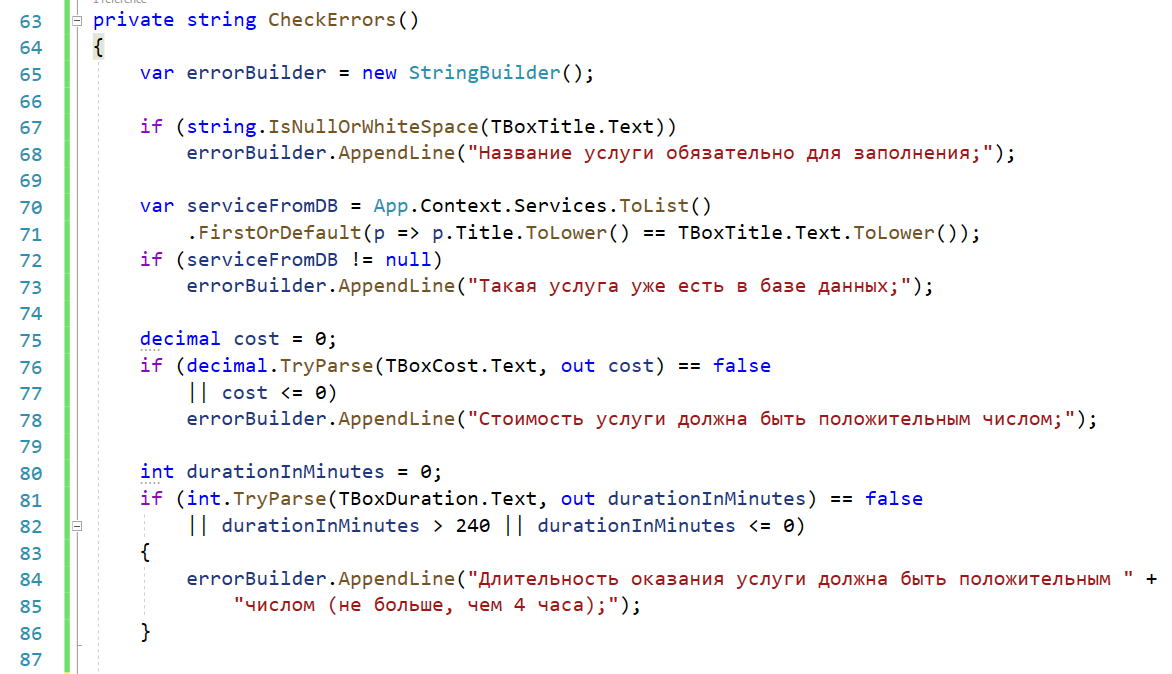


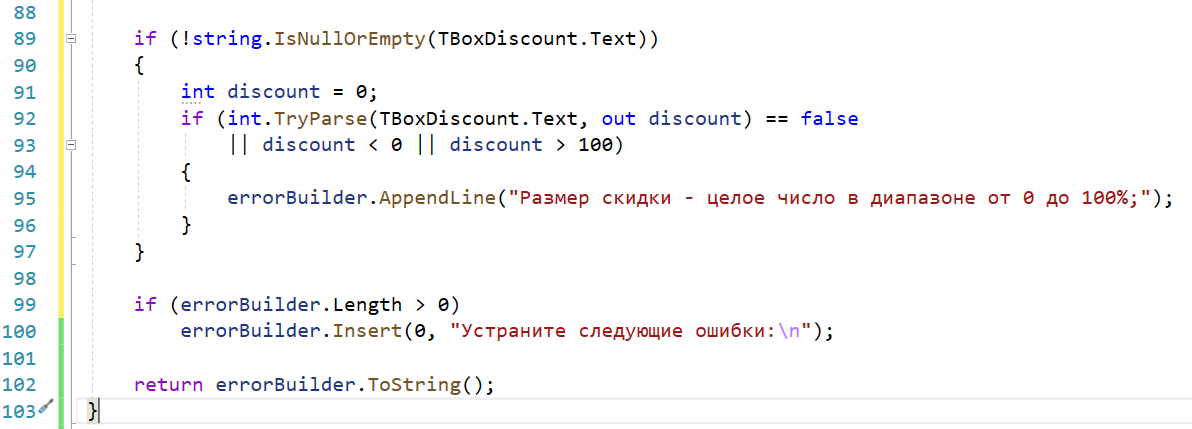
13. СЛЕД. **СТРАНИЦА РЕДАКТИРОВАНИЕ**



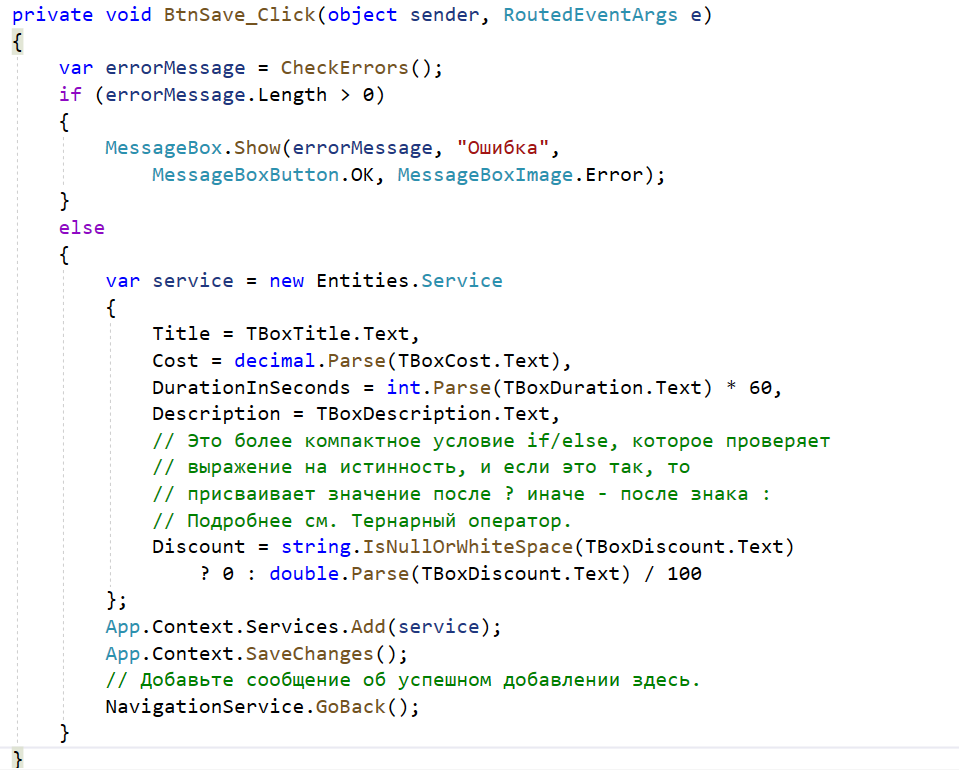


В коде страницы **добавьте метод CheckErrors** для проверки введенных значений. Этот метод будет вызываться перед добавлением/редактированием услуги – если результат выполнения метода будет пустым (сообщением с ошибками) – значит, информация введена верно и можно добавлять/редактировать услугу.

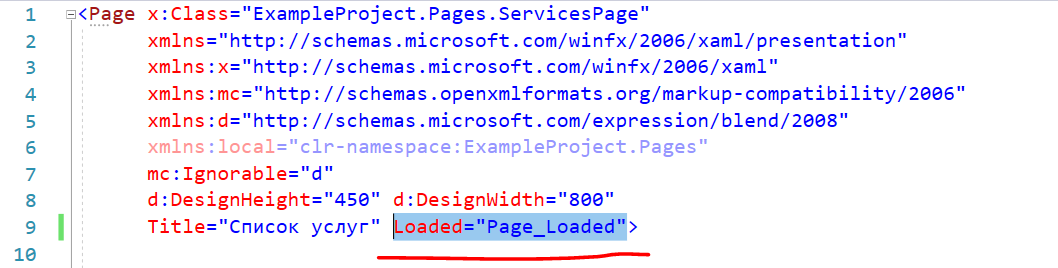


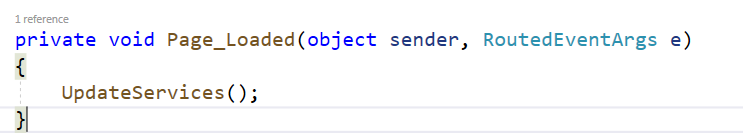


**КНОПКА «ДОБАВЛЕНИЕ»**



В странице ServicesPage обработайте событие Loaded, которое будет срабатывать каждый раз, когда страница будет загружена. Это нужно для того, чтобы услуги динамически обновлялись про добавлении/редактировании с других страниц.

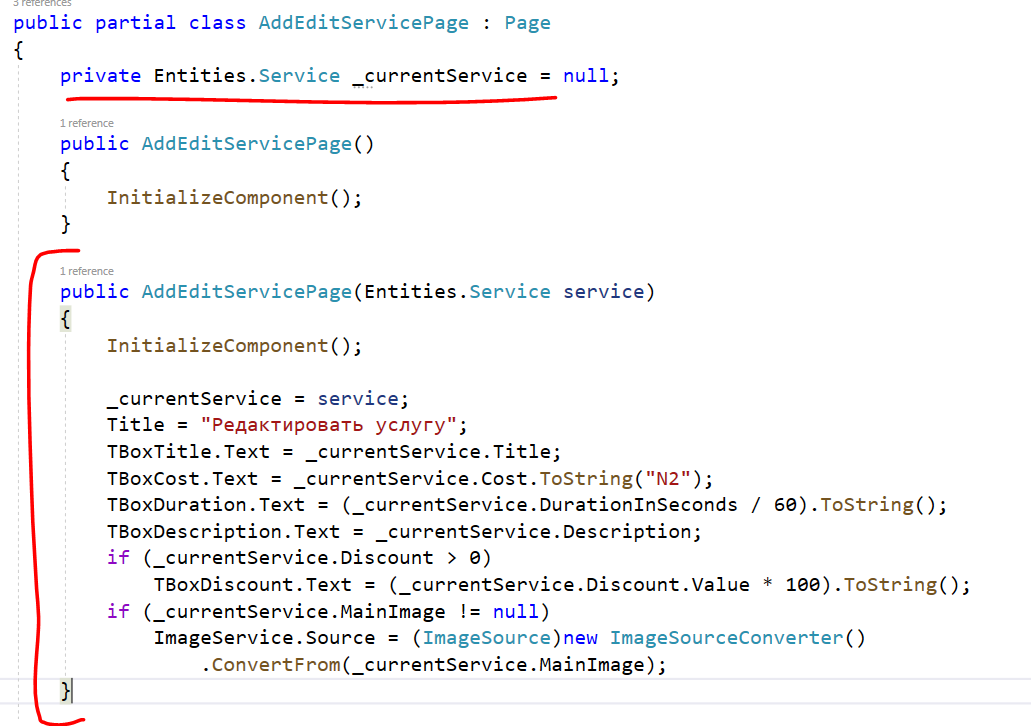




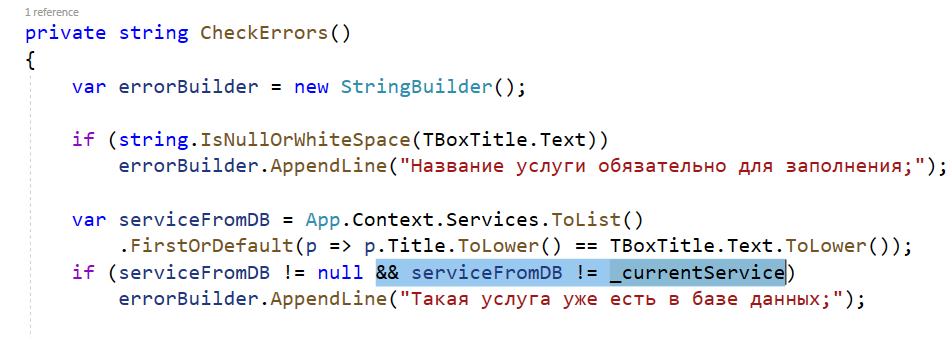
Для возможности редактирования услуги добавьте ещё один конструктор на странице AddEditServicePage(ДОБАВЛЕНИЕ УСЛУГИ)



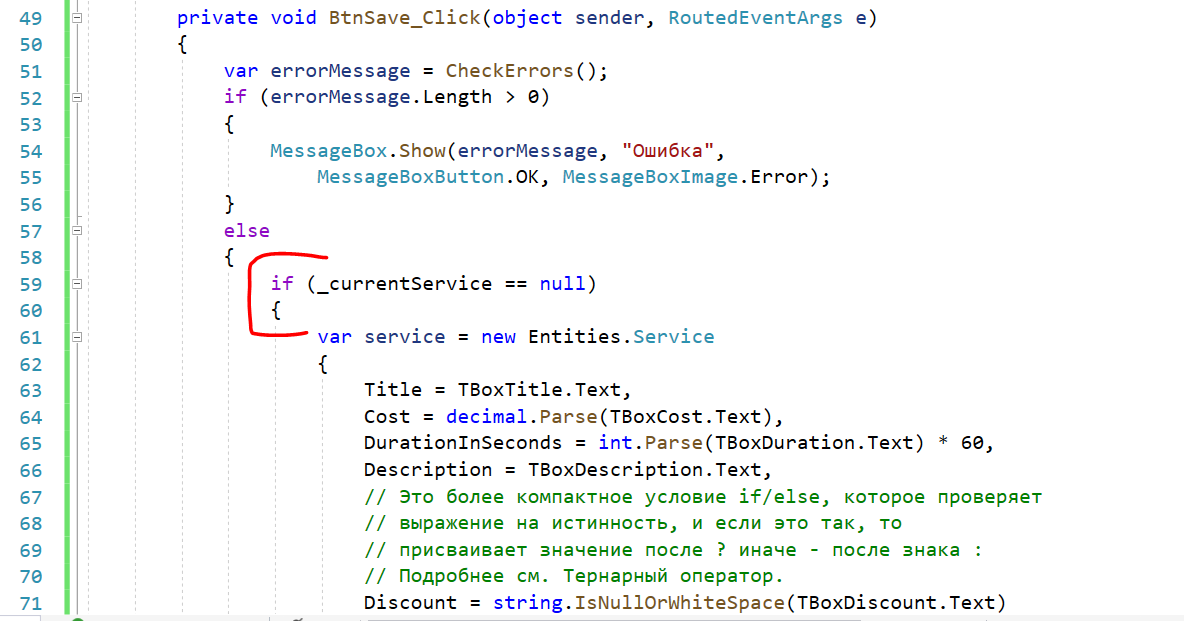
Добавьте приватное поле \_currentService, в которое будет присвоена переданная с предыдущего окна услуга. Обратите внимание, что в случае редактирования страница должна быть заполнена значениями из выбранной услуги.

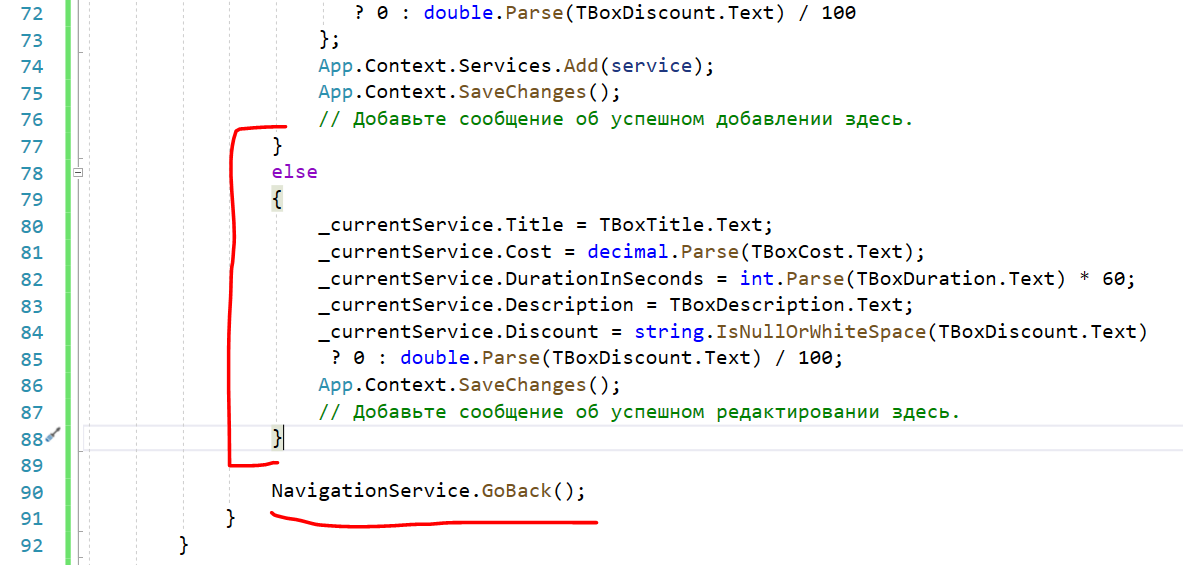


Измените логику проверки уникальности имени услуги: если найденная услуга совпадает с той, что мы редактируем, значит ошибка появляться не должна.

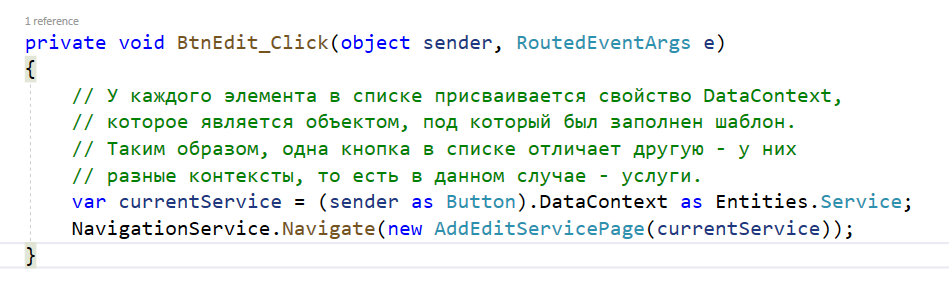


Измените код сохранения услуги, учитывая два варианта развития событий: если услуга добавляется (\_currentService == null), или редактируется (\_currentService != null). Изменения в коде выделены красной линией.





**КНОПКА РЕДАКТИРОВАТЬ**

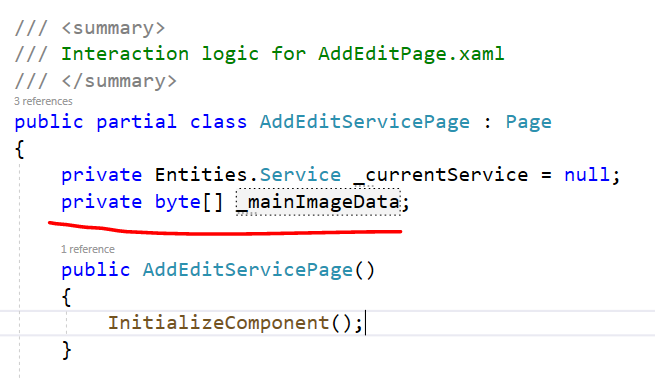


**КНОПКА УДАЛИТЬ**

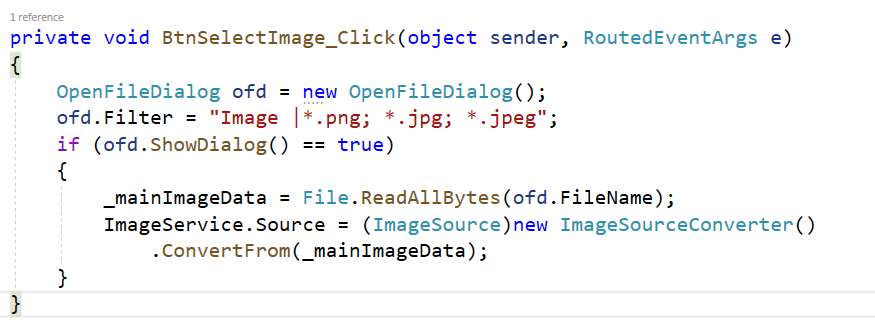


**ДОБАВЛЕНИЕ ФОТО.**

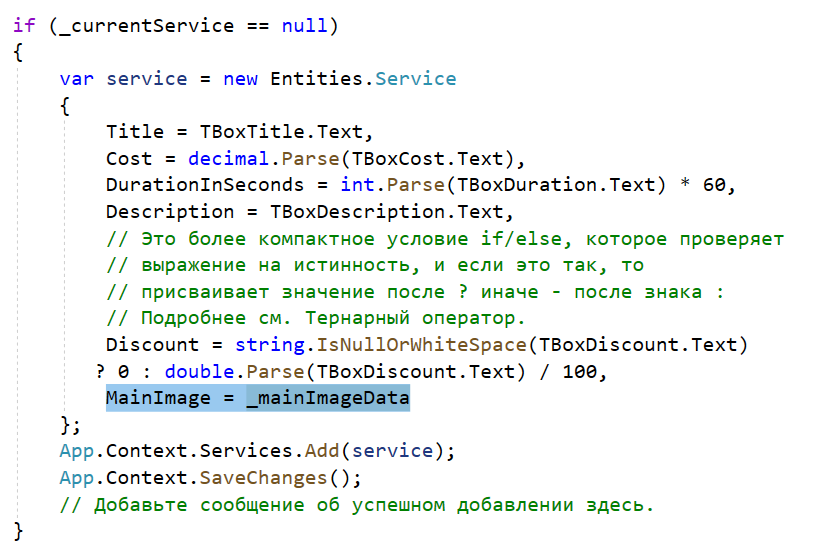
Прежде чем работать с изображением услуги, **добавьте приватное поле**, которое будет хранить массив байтов выбранного изображения (на **странице AddEditServicePage**).

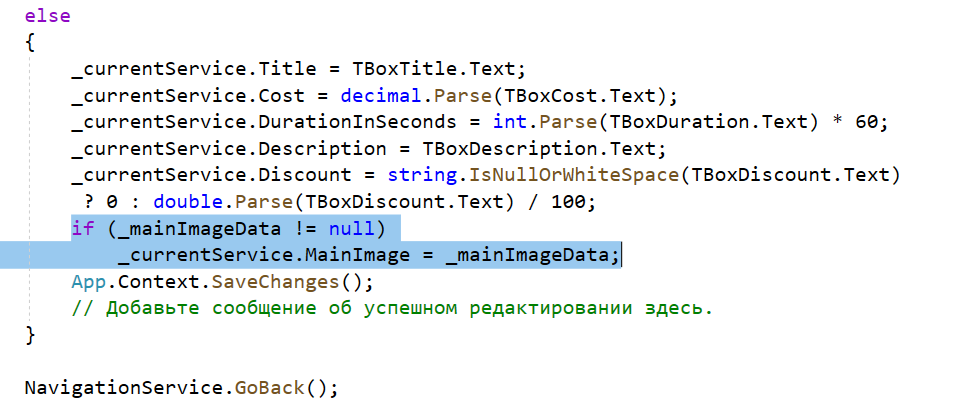


**КНОПКА «ВЫБОР ФОТОГРАФИИ»**



Осталось изменить логику добавления/редактирования (выделено красным), чтобы сохранять изображения

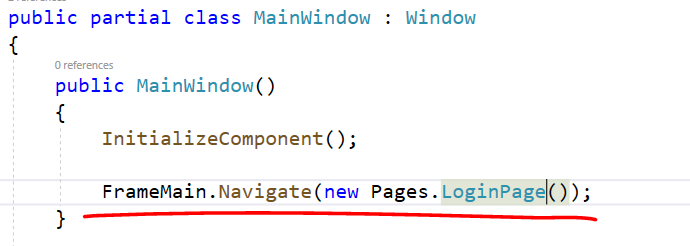




**СТРАНИЦА «АВТОРИЗАЦИЯ»**

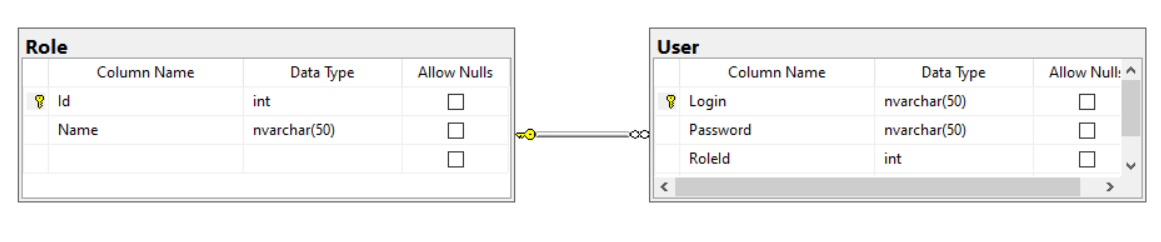


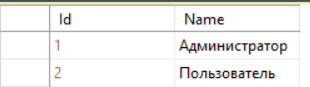
**Измените первоначальную страницу фрейма в MainWindow на LoginPage**



Прежде чем идти дальше, нам необходимо немного поменять структуру нашей базы данных. Будут добавлены таблицы User (первичный ключ - Login) и Role (первичный ключ – Id, автоинкремент).

По данным из этих таблиц мы будем распознавать пользователей и, в зависимости от их роли, предоставлять полный или урезанный функционал.





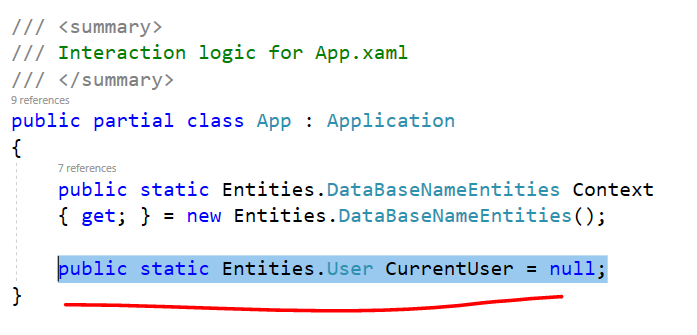
Затем необходимо обновить модель базы данных в нашем проекте. Откройте CarServiceModel двойным кликом в проекте.

Откроется примерно такое окно.

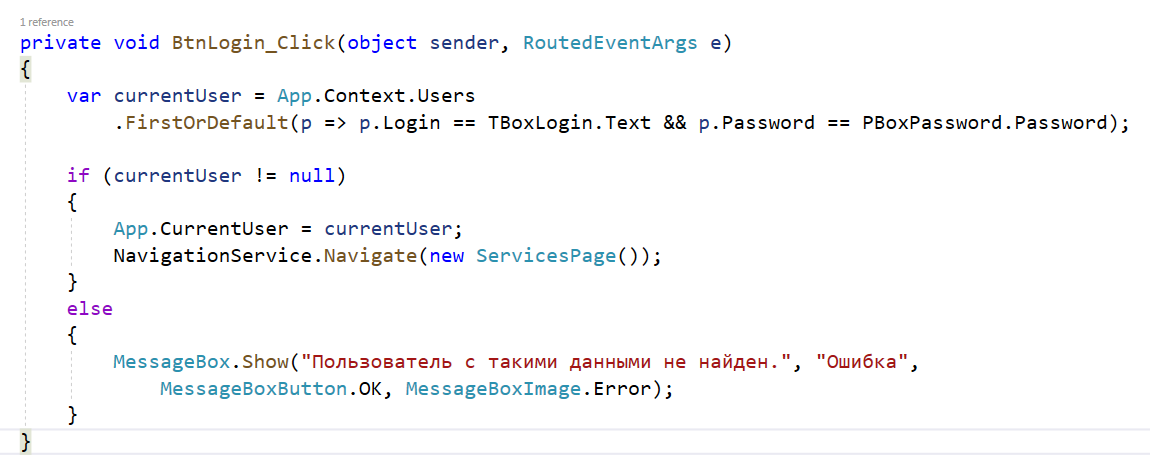
Нажмите ПКМ по свободному месту внутри окна. Выберите пункт Select All, чтобы выбрать все классы модели. Затем нажмите клавишу Delte, чтобы удалить их все.

Теперь можно приступать к обновлению модели. ПКМ в свободном месте окна, Update Model from Database…

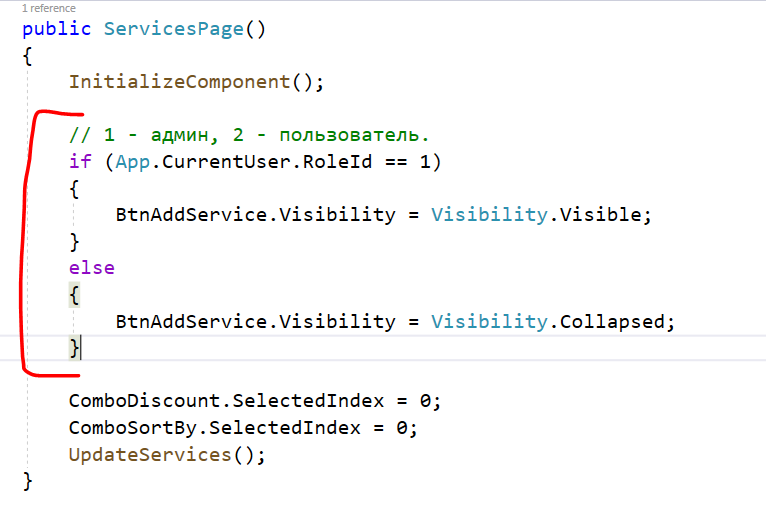
В **файл App.cs** добавьте ещё одно свойство для хранения авторизованного пользователя



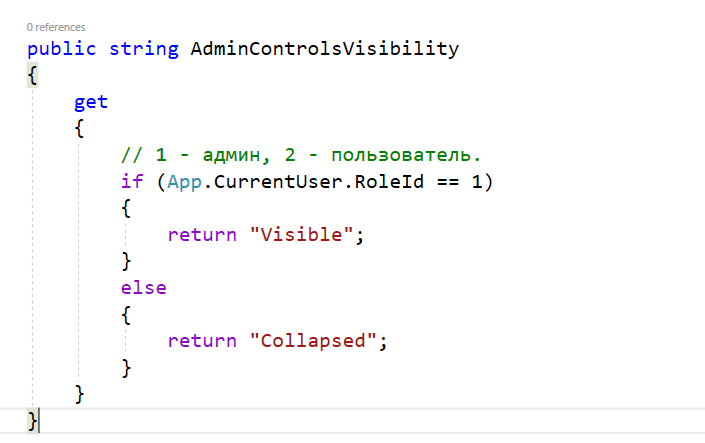
**КНОПКА АВТОРИЗАЦИИ – ВОЙТИ**



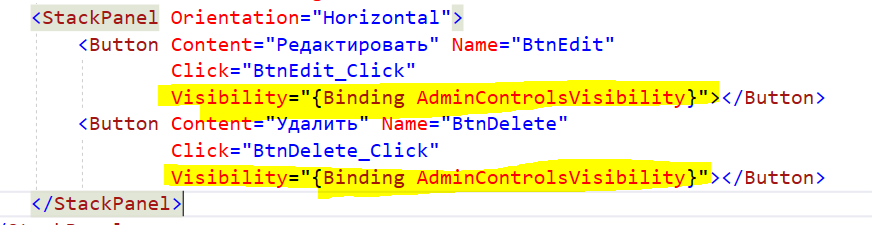
В окне **ServicesPage** пропишите новое условие для видимости кнопки добавления.



В классе ServicePartial можно добавить ещё одно свойство для определения видимости кнопок.



Осталось только привязать их видимость в верстке (в ServicesPage).



**ОТОБРАЖЕНИЕ КОЛ-ВА ЗАПИСЕЙ**

В ServicePage.xaml добавить.

Перед <ListView>

<TextBlock Grid.Row=”1” Style=”{StaticResource BlockCommon}” Width=”auto” FontWeight=”Bold” Name=”BlockRecords”></TextBlock>

В код данной страницы добавить строку в private void UpdateServices()

(последней строкой).

BlockRecords.Text = $”{Services.Count} из {TotalRecords}”;

Не забудьте добавить описание переменной в var:

int TotalRecords = services.Count;