

Laboratorium 4 Sprawozdanie z realizacji laboratorium			
Temat: Praktyczne wykorzystanie ORM + praca z GUI	Nr Albumu: 028487	Grupa/zespół: GL01	Rok/semestr: III / 6
Wykonał: Oleksii Hudzishevskiyi	Data wykonania: 15/05/2023		Data oddania: 01/07/2023
	Ocena:		Podpis prowadzącego:

1. Spis treści

1.	Spis treści.....	1
2.	Cel ćwiczenia.....	2
3.	Wymagania znajomości zagadnień	2
4.	Literatura, materiały dydaktyczne	2
5.	Wiadomości teoretyczne.....	3
6.	Przebieg ćwiczenia praktycznego.....	3
7.	Opracowanie sprawozdania	4
8.	Wnioski.....	16
9.	Bibliografia.....	16
10.	Spis ilustracji.....	16
11.	Spis snippetów	16

2. Cel ćwiczenia

Laboratorium ma na celu poszerzenie wiedzy praktycznej w zakresie wykorzystywania narzędzi ORM w trakcie tworzenia aplikacji.

3. Wymagania znajomości zagadnień

- Pisanie prostych aplikacji w C# lub innym obiektowym języku wysokiego poziomu
- Umiejętność tworzenia prostych aplikacji z wykorzystaniem GUI np. WinForms
- Umiejętność pracy na kolekcjach obiektów, proste zapytania LINQ.

4. Literatura, materiały dydaktyczne

- <https://learn.microsoft.com/pl-pl/ef/core/managing-schemas/scaffolding/?tabs=dotnet-core-cli>
- <https://www.entityframeworktutorial.net/efcore/create-model-for-existing-database-in-ef-core.aspx>
- <https://www.learnentityframeworkcore.com/walkthroughs/existing-database>
- <https://www.mdatelier.pl/entity-framework-database-first/>
- <https://www.entityframeworktutorial.net/entityframework6/dbcontext.aspx>
- <https://learn.microsoft.com/pl-pl/ef/ef6/fundamentals/working-with-dbcontext>
- <https://learn.microsoft.com/pl-pl/dotnet/api/system.windows.forms.usercontrol?view=windowsdesktop-7.0>
- <https://www.c-sharpcorner.com/UploadFile/c5c6e2/creating-a-simple-user-control-and-adding-it-to-the-toolbox/>
- <https://www.c-sharpcorner.com/article/event-handling-in-net-using-C-Sharp/>
- <https://learn.microsoft.com/en-us/dotnet/standard/events/>
- <https://www.csharptutorial.net/csharp-tutorial/csharp-events/>
- <https://dotnetcorecentral.com/blog/delegates-and-events/>

- <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/controls/how-to-access-objects-bound-to-windows-forms-datagridview-rows?view=netframeworkdesktop-4.8>
- <https://www.youtube.com/watch?v=GyLlpBZGsrE>

5. Wiadomości teoretyczne

—

6. Przebieg ćwiczenia praktycznego

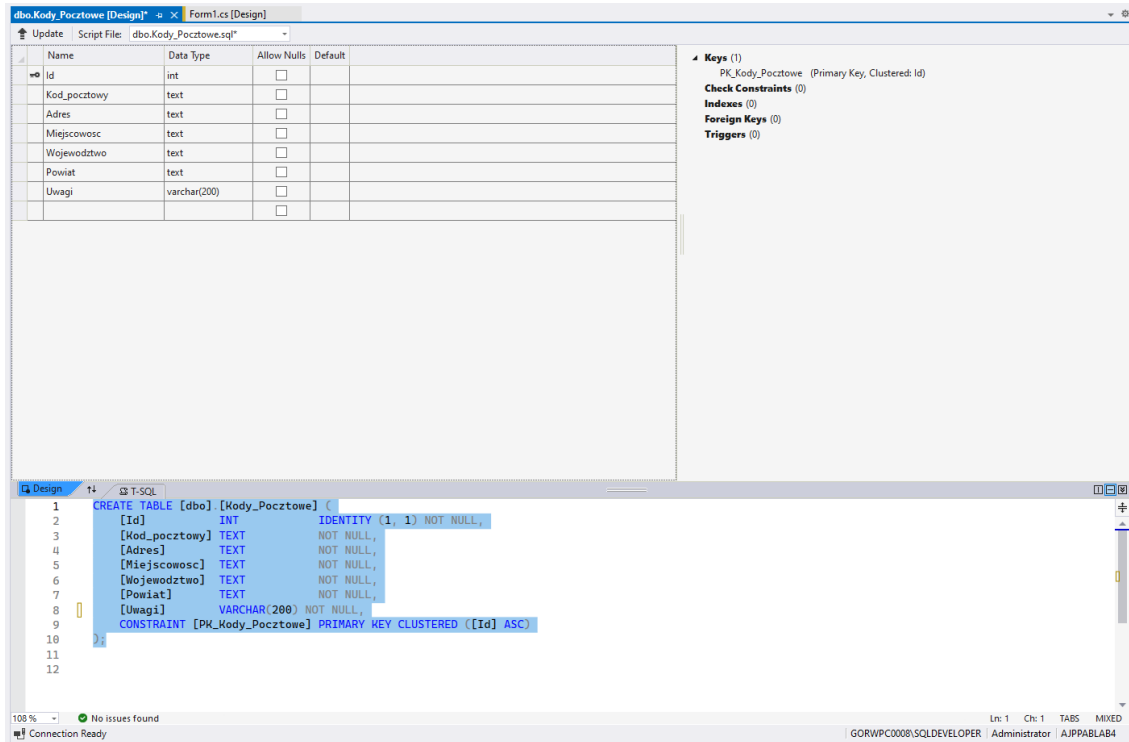
Wykorzystując danych (tabelę) wraz z danymi z poprzedniego laboratorium, zmodyfikować tabelę tak by zawierała dodatkowo 2 pola: ID, Uwagi. Id ma być kluczem głównym w tabeli z opcją autoincrement a pole uwag polem Varchar(200). Dodać tabelę lokalizacji (mogą to być sklepy, biura inne miejsca), tabela ta ma zawierać następujące pola: ID, Nazwa, Opis, oraz dodatkowo ID odnoszące się do tabeli kodów pocztowych (klucz obcy). Utworzyć projekt aplikacji WinForms, dołączyć wymagane nugety EF oraz wykorzystując polecenie Scaffold utworzyć context bazy danych wraz z modelami klas. Stworzyć kontrolkę użytkownika, która pozwalać będzie na wyświetlenie oraz edycję danych (zmiana lub kasowanie), oraz zagnieżdżony DataGridView zawierający listę wpisów powiązanych, kontrolka ta jako parametr konstruktora ma przyjmować obiekt Kodu Poczтового. Do projektu dodać formularz wyszukiwania z poprzedniego laboratorium rozbudowany o możliwość wyszukiwania również po uwagach oraz stworzyć DataGridView zawierający wszystkie spełniające kryteria wyszukiwania obiekty. Pod DataGridView dodać przycisk do edycji, jego kliknięcie ma wczytać z górnego DataGridView zaznaczoną linię (o ile jest jakaś zaznaczona) a w przypadku gdy żadna linia nie została wybrana ma wyświetlić stosowny komunikat, a następnie dynamicznie stworzyć obiekt kontrolki zawierający dany wpis, na tej kontrolce można będzie zmienić dane (nazwa ulicy, pole uwag, skasować, dodać lub edytować wpisy z tabeli lokalizacji przypisanych do danego kodu pocztowego). Kliknięcie przycisku akcji w kontrolce (kasowanie, edycja dla kodu pocztowego) powinno wyświetlić komunikat o rezultacie operacji (OK lub Error wraz z podaniem przyczyny), zamknąć kontrolkę (usunąć ją z ekranu) oraz odświeżyć dane w głównym oknie programu. W przypadku dodawania, edycji i kasowania lokalizacji przypisanych do kodu (zagnieżdżony DataGridView na kontrolce), powinno to się odbywać bezpośrednio w kontrolce. Zadbać o komunikację pomiędzy kontrolką a głównym oknem aplikacji, tak by „wiedziało ono” o tym, że wprowadzono jakiekolwiek zmiany i że lista w głównym oknie programu powinna

zostać odświeżona – w celu tym wykorzystać zdarzenia, po utworzeniu kontrolki należy przypisać w formularzu głównym zdarzenie zdefiniowane w tejże kontrolce. Przeładowanie listy na głównym oknie aplikacji ma się odbywać tylko i wyłącznie, jeżeli w trakcie pracy użytkownik wprowadzi jakiegokolwiek zmiany – kontrolka zarejestrowała fakt edycji i zakomunikowała to w stosownym zdarzeniu. Oczywiście wszystkie wprowadzane zmiany mają być wykonywane na bazie danych.

Ze względu na to, że opis interface może być niezrozumiały, wszelkie wątpliwości zostaną wyjaśnione w trakcie trwania laboratorium, na którym to, prowadzący omówi wygląd GUI oraz rozmieszczenie elementów - przy tablicy. Prowadzący przewiduje, iż wszystkie pytania oraz szczegóły implementacyjne zostaną omówione w trakcie trwania zajęć laboratoryjnych, osoby obecne w trakcie trwania zajęć będą mogły się dodatkowo skonsultować zdalnie. W przypadku osób nieobecnych na zajęciach, które uprzednio nie zgłosiły i usprawiedliwiły faktu nieobecności wsparcie to nie będzie możliwe. Możliwość konsultacji - jedynie dla studentów mających dobre usprawiedliwienie swej nieobecności. W wyjątkowym wypadku, do uznania prowadzącego, usprawiedliwienie nieobecności może nastąpić po zajęciach, ale zostanie zaakceptowane jedynie wtedy, gdy będzie naprawdę zasadne. Informuję również, iż przez wzgląd na notoryczne zaniżanie frekwencji i niestawianie się na zajęciach, poziom kolejnych zadań laboratoryjnych będzie zdecydowanie zawyżony, przy podniesionej punktacji oraz przy konsultacjach zarezerwowanych jedynie dla osób obecnych na zajęciach lub też posiadających stosowne usprawiedliwienie swej nieobecności. Do sprawozdania należy **OBOWIĄZKOWO** dołączyć spakowany projekt aplikacji, zawierający wszystkie kody źródłowe.

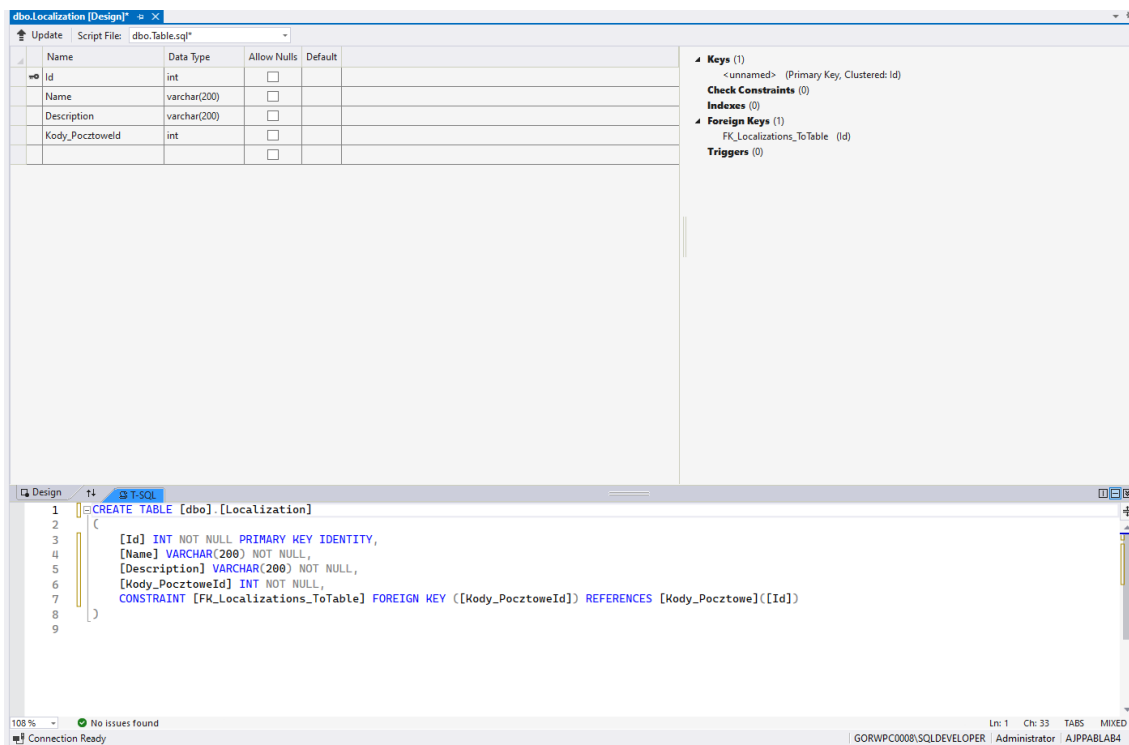
7. Opracowanie sprawozdania

Wykorzystując tabele wraz z danymi z poprzedniego laboratorium zmodyfikowane zostały tabele o dwa dodatkowe pola: **Id**, **Uwagi**. Id jest kluczem głównym w tabeli z opcją autoincrement a pole uwag polem Varchar(200). Schemat tej tabeli jest pokazany na *Rysunek 1*



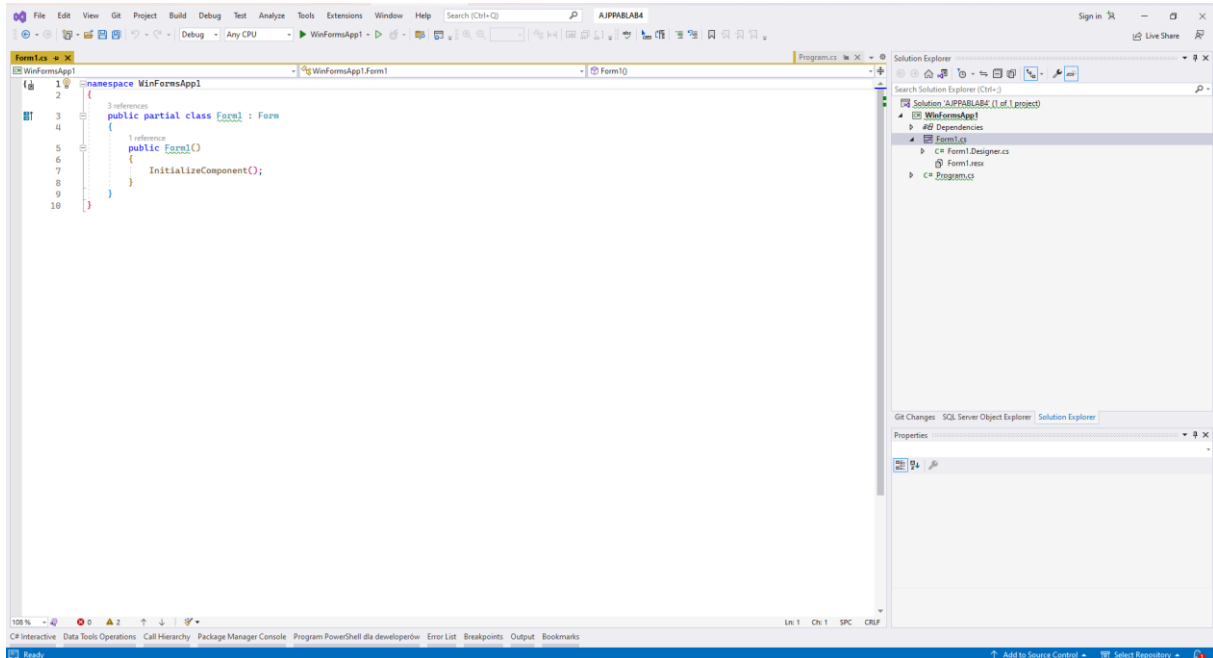
Rysunek 1. Schemat tabeli Kody_Pocztowe

Dodatkowo została tabela o nazwie Localization, tabela ta zawiera następujące pola: Id, Name, Description oraz Kody_PocztoweId (klucz obcy). Schemat tej tabeli jest pokazany na



Rysunek 2. Schemat tabeli Localization

Utworzony został projekt Window Forms .NET 7 i do niego zostały dołączone pakiety z Entity Framework’iem. Wygląd projektu oraz jego struktura pokazane na *Rysunek 3* , a skrypt, który umożliwia zainstalowanie pakietów Nu-Get pokazany na *Snippet 1*.



Rysunek 3. Struktura projektu

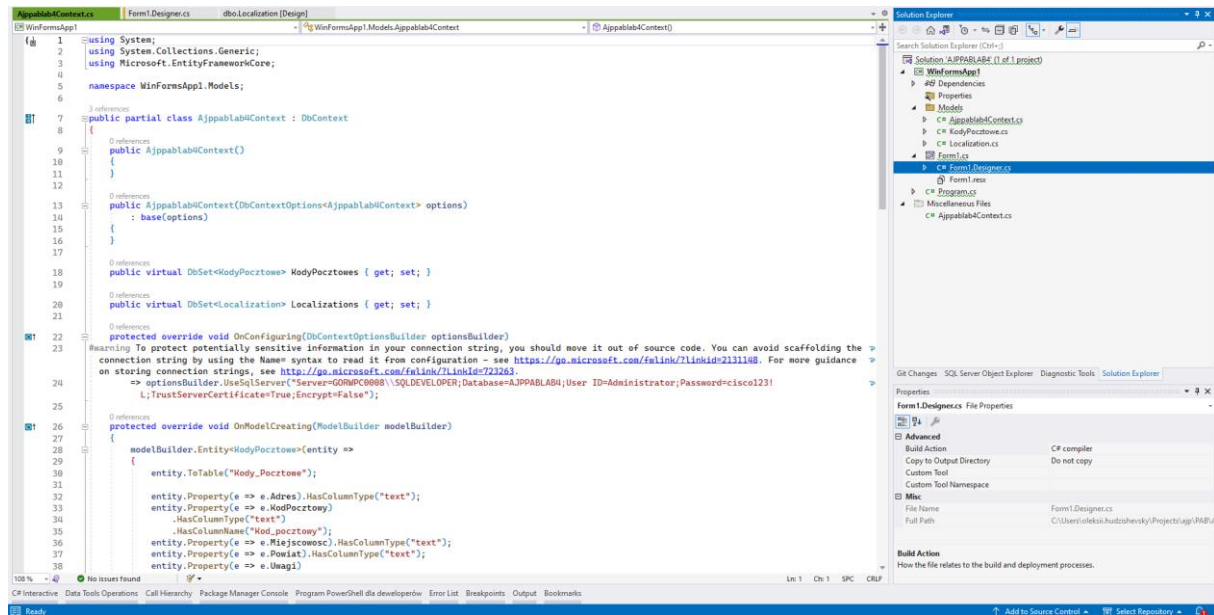
Snippet 1. Skrypt do zainstalowania EF

```
Install-Package Microsoft.EntityFrameworkCore
Install-Package Microsoft.EntityFrameworkCore.Relational
Install-Package Microsoft.EntityFrameworkCore.Abstractions
Install-Package Microsoft.EntityFrameworkCore.Design
Install-Package Microsoft.EntityFrameworkCore.Tools
Install-Package Microsoft.EntityFrameworkCore.Proxies
Install-Package Microsoft.EntityFrameworkCore.SqlServer
```

Następnie za pomocą polecenia Scaffold utworzony został kontekst bazy danych wraz z tabelami w postaci klas. Skrypt wykonujący generację tego kodu zaprezentowany jest na *Snippet 2*, a sam wygenerowany kod wraz z plikami pokazany na *Rysunek 4*.

Snippet 2. Polecenie Scaffold-DbContext

```
Scaffold-DbContext "Server=GORWPC0008\SQLDEVELOPER;Database=AJPPA-BLAB4;User ID=Administrator;Password=cisco123!L;TrustServerCertificate=True;Encrypt=False" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models
```



Rysunek 4. Kontekst bazy danych

Następnie w pliku Form1 został utworzony widok z kontrolkami do zrealizowania laboratorium. Widok aplikacji jest pokazany *Rysunek 5*. Widok programu zawiera:

- DataGridView do wyświetlania tabeli Kody_Pocztowe z bazy danych,
- DataGridView do wyświetlania tabeli Localization z bazy danych,
- DataGridView, który wyświetla wynik wyszukiwania z tabeli Kody_Pocztowe
- DataGridView, który wyświetla powiązane dane z wyświetlanymi na innym DataGridView (z danymi Kody_Pocztowe)
- SearchBox zawiera TextBox'y z kryteriami do wyszukiwania konkretnych informacji w tabeli Kody_Pocztowe,
- Kod_Pocztowy Change Box zawiera formularz, w którym możliwa jest modyfikacja wybranego rekordu z tabeli Kody_Pocztowe lub utworzenie nowego rekordu. Jest ukryty, dopóki nie zostanie aktywowany przyciskiem „Edit Kod_Pocztowy”,
- Localization Change Box zawiera formularz, w którym możliwa jest modyfikacja wybranego rekordu z tabeli Localization. Jest ukryty, dopóki nie zostanie aktywowany przyciskiem „Edit Localization,

Akademia im. Jakuba z Paradyża w Gorzowie Wielkopolskim

Wydział Techniczny

The screenshot shows a web application window titled 'Form1'. It contains two main data grids and a search form. The top-left grid, 'Data from Database', lists addresses in Warsaw. The top-right grid, 'Data from Database', lists shops. The bottom-left grid, 'Search Result', shows a list of addresses. The bottom-right grid, 'Search Result', shows a list of shops. The search form in the center has fields for Id, Kod pocztowy, Adres, Mięscowosc, Województwo, Powiat, and Uwagi, along with buttons for Search, Edit Kod pocztowy, and Edit Localization.

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
2	00-001	Pocztowa Warszawa	Warszawa	Województwo ...	Warszawa	
3	00-001	ul. Świętokrzyska	Warszawa	Województwo ...	Warszawa	
4	00-002	ul. Świętokrzyska	Warszawa	Województwo ...	Warszawa	
5	00-003	ul. Jasna od 9 d...	Warszawa	Województwo ...	Warszawa	
6	00-004	ul. Marszałkowsk...	Warszawa	Województwo ...	Warszawa	
7	00-005	ul. Rysia od 1 d...	Warszawa	Województwo ...	Warszawa	

Id	Name	Description	KodyPocztoweId
3	Sklep	Sklep z materiał...	2
4	Galeria	Miejsce rozryw...	14
5	Silownia	Więcej, szybciej...	124

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
120	00-112	ul. Bagno	Warszawa	Województwo ...	Warszawa	
121	00-113	ul. Plater Emili...	Warszawa	Województwo ...	Warszawa	
122	00-114	ul. Twarda od 1 ...	Warszawa	Województwo ...	Warszawa	
123	00-115	Pl. Grzybowski ...	Warszawa	Województwo ...	Warszawa	
124	00-116	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	
125	00-116	ul. Świętokrzysk...	Warszawa	Województwo ...	Warszawa	
126	00-117	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	
127	00-117	ul. Chmielna o...	Warszawa	Województwo ...	Warszawa	
128	00-118	ul. Plater Emili...	Warszawa	Województwo ...	Warszawa	
129	00-119	ul. Sosnowa	Warszawa	Województwo ...	Warszawa	

Id	Name	Description	KodyPocztoweId
3	Sklep	Sklep z materiał...	2
5	Silownia	Więcej, szybciej...	124

Rysunek 5. Widok aplikacji

Po wpisaniu wyszukiwanego kryterium i kliknięciu przyciska „Search” w dolnej części zostaną wyświetlone Kody_Pocztowe i w drugim dolnym DataGridView zostaną wyświetlone powiązane z nim lokalizacje. Przykładowe działanie pokazane jest na *Rysunek 6*. Kod realizujący tę funkcjonalność jest pokazany na *Snippet 3*.

The screenshot shows a web application window titled 'Form1'. It contains two main data grids and a search form. The top-left grid, 'Data from Database', lists addresses in Warsaw. The top-right grid, 'Data from Database', lists shops. The bottom-left grid, 'Search Result', shows a list of addresses. The bottom-right grid, 'Search Result', shows a list of shops. The search form in the center has fields for Id, Kod pocztowy, Adres, Mięscowosc, Województwo, Powiat, and Uwagi, along with buttons for Search, Edit Kod pocztowy, and Edit Localization.

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
2	00-001	Pocztowa Warszawa	Warszawa	Województwo ...	Warszawa	Nowe uwagi 3
3	00-001	ul. Świętokrzyska	Warszawa	Województwo ...	Warszawa	
4	00-002	ul. Świętokrzyska	Warszawa	Województwo ...	Warszawa	
5	00-003	ul. Jasna od 9 d...	Warszawa	Województwo ...	Warszawa	
6	00-004	ul. Marszałkowsk...	Warszawa	Województwo ...	Warszawa	
7	00-005	ul. Rysia od 1 d...	Warszawa	Województwo ...	Warszawa	

Id	Name	Description	KodyPocztoweId
3	Sklep Tesco	Sklep z materiał...	2
4	Galeria	Miejsce rozryw...	14
5	Silownia	Więcej, szybciej...	124

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
124	00-116	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	
1124	01-113	ul. Ułtychowska	Warszawa	Województwo ...	Warszawa	
1240	01-210	ul. Kolejowa od...	Warszawa	Województwo ...	Warszawa	
1241	01-210	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
1242	01-211	ul. Giełdowa	Warszawa	Województwo ...	Warszawa	
1243	01-211	ul. Kasprzaka M...	Warszawa	Województwo ...	Warszawa	
1244	01-212	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
1245	01-213	ul. Korczaka Jan...	Warszawa	Województwo ...	Warszawa	
1246	01-213	ul. Szymańskie...	Warszawa	Województwo ...	Warszawa	
1247	01-214	ul. Laskowa	Warszawa	Województwo ...	Warszawa	

Id	Name	Description	KodyPocztoweId
5	Silownia	Więcej, szybciej...	124

Rysunek 6. Wyświetlanie powiązanych danych

Po zaznaczeniu wiersza z kodem pocztowym i kliknięciu przyciska „Edit Kod Pocz-
towy” zostanie wyświetlony dodatkowy formularz z wczytanymi danymi z zaznaczonego wiersza, jeśli żaden wiersz nie był zaznaczony – zostanie wyświetlony pusty formularz, w którym można dodać nowy rekord. Przykładowe działanie pokazane jest na *Rysunek 7*. Po kliknięciu przyciska „Delete Record” zaznaczony wiersz zostanie usunięty. Kod realizujący te funkcjonalności pokazany jest na *Snippet 4* i *Snippet 5* i *Snippet 6*.

The screenshot shows a software application window titled "Form1". It contains two data tables and two forms.

Data from Database (Left Table):

	Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
▶	2	00-001	Pocztowa Warsz...	Warszawa	Województwo ...	Warszawa	Nowe uwagi 3
	3	00-001	ul. Świętokrzysk...	Warszawa	Województwo ...	Warszawa	
	4	00-002	ul. Świętokrzysk...	Warszawa	Województwo ...	Warszawa	
	5	00-003	ul. Jasna od 9 d...	Warszawa	Województwo ...	Warszawa	
	6	00-004	ul. Marszałkows...	Warszawa	Województwo ...	Warszawa	
	7	00-005	ul. Rysia od 1 d...	Warszawa	Województwo ...	Warszawa	

Data from Database (Right Table):

	Id	Name	Description	KodyPocztoweld
▶	3	Sklep Tesco	Sklep z materiał...	2
	4	Galeria	Miejsce rozyw...	14
	5	Silownia	Więcej, szybciej...	124

Search Box (Left Form):

Id: 124, Miejscowosc: [empty], Kod pocztowy: [empty], Wojewodztwo: [empty], Adres: [empty], Powiat: [empty], Uwagi: [empty]. Buttons: Search, Edit Kod pocztowy, Edit Localization.

Kod Pocztowy Change Box (Right Form):

Id: 124, Miejscowosc: Warszawa, Kod Pocztowy: 00-116, Wojewodztwo: Województwo mazowieckie, Adres: Al. Jana Pawła II od 18 do 18, Powiat: Warszawa, Uwagi: Nowe Uwagi. Buttons: Apply Changes, Delete Record.

Search Result (Left Table):

	Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
▶	124	00-116	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	
	1124	01-113	ul. Ulrychowska	Warszawa	Województwo ...	Warszawa	
	1240	01-210	ul. Kolejowa od...	Warszawa	Województwo ...	Warszawa	
	1241	01-210	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
	1242	01-211	ul. Gieldowa	Warszawa	Województwo ...	Warszawa	
	1243	01-211	ul. Kasprzaka M...	Warszawa	Województwo ...	Warszawa	
	1244	01-212	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
	1245	01-213	ul. Korczaka Jan...	Warszawa	Województwo ...	Warszawa	
	1246	01-213	ul. Szymańskie...	Warszawa	Województwo ...	Warszawa	
	1247	01-214	ul. Laskowa	Warszawa	Województwo ...	Warszawa	

Search Result (Right Table):

	Id	Name	Description	KodyPocztoweld
▶	5	Silownia	Więcej, szybciej...	124

Rysunek 7. Modyfikacja kodu pocztowego

Po zaznaczeniu wiersza z lokalizacją i kliknięciu przyciska „Edit Localization” zostanie wyświetlony dodatkowy formularz z wczytanymi danymi z zaznaczonego wiersza, jeśli żaden wiersz nie był zaznaczony – zostanie wyświetlony pusty formularz, w którym można dodać nowy rekord. Przykładowe działanie pokazane jest na *Rysunek 8*. Po kliknięciu przyciska „Delete Record” zaznaczony wiersz zostanie usunięty. Kod realizujący te funkcjonalności pokazany jest na *Snippet 7* i *Snippet 8* i *Snippet 9*.

Data from Database

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
122	00-114	ul. Twarda od 1 ...	Warszawa	Województwo ...	Warszawa	
123	00-115	Pl. Grzybowski ...	Warszawa	Województwo ...	Warszawa	
124	00-116	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	Nowe Uwagi
125	00-116	ul. Świętokrzysk...	Warszawa	Województwo ...	Warszawa	
126	00-117	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	
127	00-117	ul. Chmielna o...	Warszawa	Województwo ...	Warszawa	

Data from Database

Id	Name	Description	KodyPocztoweld
3	Sklep Tesco	Sklep z materiał...	2
4	Galeria	Miejsce rozyw...	14
5	Silownia	Wiecej, szybciej...	124

Search Box

Id: 124, Miejscowosc:

Kod pocztowy: , Wojewodztwo:

Adres: , Powiat:

Uwagi:

Buttons: Search, Edit Kod pocztowy, Edit Localization

Localization Change Box

Id: 5, Kod pocztowy Id: 124

Name: Silownia

Description: Wiecej, szybciej, mocniej

Buttons: Apply Changes, Delete Record

Search Result

Id	KodPocztowy	Adres	Miejscowosc	Wojewodztwo	Powiat	Uwagi
124	00-116	Al. Jana Pawła II...	Warszawa	Województwo ...	Warszawa	Nowe Uwagi
1124	01-113	ul. Ulrychowska	Warszawa	Województwo ...	Warszawa	
1240	01-210	ul. Kolejowa od...	Warszawa	Województwo ...	Warszawa	
1241	01-210	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
1242	01-211	ul. Giełdowa	Warszawa	Województwo ...	Warszawa	
1243	01-211	ul. Kasprzaka M...	Warszawa	Województwo ...	Warszawa	
1244	01-212	ul. Szarych Szer...	Warszawa	Województwo ...	Warszawa	
1245	01-213	ul. Korczaka Jan...	Warszawa	Województwo ...	Warszawa	
1246	01-213	ul. Szymańskie...	Warszawa	Województwo ...	Warszawa	
1247	01-214	ul. Laskowa	Warszawa	Województwo ...	Warszawa	

Search Result

Id	Name	Description	KodyPocztoweld
5	Silownia	Wiecej, szybciej...	124

Rysunek 8. Modyfikacja lokalizacji

Snippet 3. Search Button

```
private void searchButton_Click(object sender, EventArgs e)
{
    try
    {
        string? id = idTextBox.Text.Length > 0 ? idTextBox.Text : null;
        string? miejscowosc = miejscowoscTextBox.Text.Length > 0 ? miejscowoscTextBox.Text :
null;
        string? powiat = powiatTextBox.Text.Length > 0 ? powiatTextBox.Text : null;
        string? adres = adresTextBox.Text.Length > 0 ? adresTextBox.Text : null;
        string? kodPocztowy = kodPocztowyTextBox.Text.Length > 0 ? kodPocztowyTextBox.Text :
null;
        string? wojewodztwo = wojewodztwoTextBox.Text.Length > 0 ? wojewodztwoTextBox.Text :
null;
        string? uwagi = uwagiTextBox.Text.Length > 0 ? uwagiTextBox.Text : null;

        List<KodyPocztowe> result = new();
        List<Localization> localizationsResult = new();
        Stopwatch stopwatch = new();
        stopwatch.Start();
        using (Ajppablab4Context db = new())
        {
            result = db.KodyPocztowes.ToList();
            if (id != null)
            {
                result = result.Where(kod => kod.Id.ToString().ToLower().Contains(id.ToLo-
wer())).ToList();
            }
            if (miejscowosc != null)
            {
                result = result.Where(kod => kod.Miejscowosc.ToLower().Contains(miejscow-
osc.ToLower())).ToList();
            }
        }
    }
}
```

```
        }
        if (powiat != null)
        {
            result = result.Where(kod => kod.Powiat.ToLower().Contains(powiat.ToLo-
wer())).ToList();
        }
        if (adres != null)
        {
            result = result.Where(kod => kod.Adres.ToLower().Contains(adres.ToLo-
wer())).ToList();
        }
        if (kodPocztowy != null)
        {
            result = result.Where(kod => kod.KodPocztowy.ToLower().Contains(kodPocz-
towy.ToLower())).ToList();
        }
        if (województwo != null)
        {
            result = result.Where(kod => kod.Wojewodztwo.ToLower().Contains(wojewodz-
two.ToLower())).ToList();
        }
        if (województwo != null)
        {
            result = result.Where(kod => kod.Uwagi.ToLower().Contains(uwagi.ToLo-
wer())).ToList();
        }
        if (result.Count < 0)
        {
            throw new Exception("Nothing Found");
        }

        if (result.Count > 0)
        {
            foreach (var item in result)
            {
                localizationsResult.AddRange(db.Localizations.Where(l => l.KodyPocztoweId
== item.Id).ToList());
            }
        }
        stopwatch.Stop();

        kodyPocztoweSearchedBindingSource.DataSource = result;
        localizationSearchedBindingSource.DataSource = localizationsResult;

        MessageBox.Show($"Search Result: {result.Count} items \nIn time: {stopwatch.Elap-
sed}");
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Occured error {ex.Message}");
    }
}
```

Snippet 4. Edit Kod_Pocztowy Button

```
private void editKodPocztowyButton_Click(object sender, EventArgs e)
{
    try
    {
        kodPocztowyChangeGroupBox.Visible = true;
        var selectedRows = kodPocztowySearchedGridView.SelectedRows;

        if (selectedRows.Count > 0)
        {
            var selectedRow = selectedRows[0];

            changeKodPocztowyIdTextBox.Text = selectedRow.Cells[0].Value.ToString();
            changeKodPocztowyTextBox.Text = selectedRow.Cells[1].Value.ToString();
            changeKodPocztowyAdresTextBox.Text = selectedRow.Cells[2].Value.ToString();
            changeKodPocztowyMiejscowoscTextBox.Text = selectedRow.Cells[3].Value.ToString();
            changeKodPocztowyWojewodztwoTextBox.Text = selectedRow.Cells[4].Value.ToString();
        }
    }
}
```

```
        changeKodPocztowyPowiatTextBox.Text = selectedRow.Cells[5].Value.ToString();
        changeKodPocztowyUwagiTextBox.Text = selectedRow.Cells[6].Value.ToString();
    }
    else
    {
        MessageBox.Show($"You have not selected any row to modify, and now a form to create a new object is displayed");
    }
}
catch (Exception ex)
{
    MessageBox.Show($"Occured error {ex.Message}");
    kodPocztowyChangeGroupBox.Visible = false;
}
}
```

Snippet 5. Change Kod_Pocztowy Button

```
private void changesKodPocztowyButton_Click(object sender, EventArgs e)
{
    try
    {
        if (changeKodPocztowyIdTextBox.Text == "")
        {
            using (Ajppablab4Context db = new())
            {
                KodyPocztowe newKodPocztowy = new()
                {
                    KodPocztowy = changeKodPocztowyTextBox.Text,
                    Miejscowosc = changeKodPocztowyMiejscowoscTextBox.Text,
                    Wojewodztwo = changeKodPocztowyWojewodztwoTextBox.Text,
                    Adres = changeKodPocztowyAdresTextBox.Text,
                    Powiat = changeKodPocztowyPowiatTextBox.Text,
                    Uwagi = changeKodPocztowyUwagiTextBox.Text,
                };

                db.KodyPocztowes.Add(newKodPocztowy);
                db.SaveChanges();
            }
        }
        else
        {
            int? id = null;
            if (Int32.TryParse(changeKodPocztowyIdTextBox.Text, out _)) id = Int32.Parse(changeKodPocztowyIdTextBox.Text);

            using (Ajppablab4Context db = new())
            {
                var updatedKodPocztowy = db.KodyPocztowes.Where(kp => kp.Id == id).FirstOrDefault();

                updatedKodPocztowy.KodPocztowy = changeKodPocztowyTextBox.Text;
                updatedKodPocztowy.Miejscowosc = changeKodPocztowyMiejscowoscTextBox.Text;
                updatedKodPocztowy.Wojewodztwo = changeKodPocztowyWojewodztwoTextBox.Text;
                updatedKodPocztowy.Adres = changeKodPocztowyAdresTextBox.Text;
                updatedKodPocztowy.Powiat = changeKodPocztowyPowiatTextBox.Text;
                updatedKodPocztowy.Uwagi = changeKodPocztowyUwagiTextBox.Text;

                db.SaveChanges();
            }
        }

        clearKodyResultDataGridView();
        clearLocalizationsResultDataGridView();
        fillKodyDataGridView();
        fillLocalizationsDataGridView();
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Occured error {ex.Message}");
    }
    finally
    {
    }
}
```

```
{
    changeKodPocztowyTextBox.Text = "";
    changeKodPocztowyMiejscowoscTextBox.Text = "";
    changeKodPocztowyWojewodztwoTextBox.Text = "";
    changeKodPocztowyAdresTextBox.Text = "";
    changeKodPocztowyPowiatTextBox.Text = "";
    changeKodPocztowyUwagiTextBox.Text = "";
    kodPocztowyChangeGroupBox.Visible = false;
}
}
```

Snippet 6. Delet Kod_Pocztowy Button

```
private void deleteKodPocztowyButton_Click(object sender, EventArgs e)
{
    try
    {
        int? id = null;
        if (Int32.TryParse(changeKodPocztowyIdTextBox.Text, out _)) id = Int32.Parse(change-
KodPocztowyIdTextBox.Text);

        using (Ajppablab4Context db = new())
        {
            var deletedKodPocztowy = db.KodyPocztowes.Where(kp => kp.Id == id).FirstOrDe-
fault();
            db.KodyPocztowes.Remove(deletedKodPocztowy);
            db.SaveChanges();
        }

        clearKodyResultDataGridView();
        clearLocalizationsResultDataGridView();
        fillKodyDataGridView();
        fillLocalizationsDataGridView();
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Occured error {ex.Message}");
    }
    finally
    {
        changeKodPocztowyTextBox.Text = "";
        changeKodPocztowyMiejscowoscTextBox.Text = "";
        changeKodPocztowyWojewodztwoTextBox.Text = "";
        changeKodPocztowyAdresTextBox.Text = "";
        changeKodPocztowyPowiatTextBox.Text = "";
        changeKodPocztowyUwagiTextBox.Text = "";
        kodPocztowyChangeGroupBox.Visible = false;
    }
}
```

Snippet 7. Edit Localization Button

```
private void editLocalizationButton_Click(object sender, EventArgs e)
{
    try
    {
        localizationChangeGroupBox.Visible = true;
        var selectedRows = localizationsSearchedGridView.SelectedRows;

        if (selectedRows.Count > 0)
        {
            var selectedRow = selectedRows[0];

            changeLocalizationIdTextBox.Text = selectedRow.Cells[0].Value.ToString();
            changeLocalizationKodPocztowyIdTextBox.Text = selectedRow.Cells[3].Value.To-
String();
            changeLocalizationNameTextBox.Text = selectedRow.Cells[1].Value.ToString();
            changeLocalizationDescriptionTextBox.Text = selectedRow.Cells[2].Value.ToString();
        }
        else
    }
```

```
        {
            MessageBox.Show($"You have not selected any row to modify, and now a form to create a new object is displayed");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Occured error {ex.Message}");
        localizationChangeGroupBox.Visible = true;
    }
}
```

Snippet 8. Changes Localization Button

```
private void changesLocalizationButton_Click(Object sender, EventArgs e)
{
    try
    {
        if (changeLocalizationIdTextBox.Text == "")
        {
            using (Ajppablab4Context db = new())
            {
                int id;
                if (Int32.TryParse(changeLocalizationKodPocztowyIdTextBox.Text, out _))
                {
                    id = Int32.Parse(changeLocalizationKodPocztowyIdTextBox.Text);
                    var foreingKodPocztowy = db.KodyPocztowes.Where(kp => kp.Id == id).FirstOrDefault();

                    if (foreingKodPocztowy == null)
                    {
                        MessageBox.Show($"You are trying to bind a new object to a non-existent 'Kode Pocztowy'");
                    }
                    else
                    {
                        Localization newLocalization = new()
                        {
                            KodyPocztoweId = id,
                            Name = changeLocalizationNameTextBox.Text,
                            Description = changeLocalizationDescriptionTextBox.Text
                        };

                        db.Localizations.Add(newLocalization);
                        db.SaveChanges();
                    }
                }
                else
                {
                    MessageBox.Show($"'Kod Pocztowy Id' must be Integer");
                }
            }
        }
        else
        {
            int id = Int32.Parse(changeLocalizationIdTextBox.Text);

            using (Ajppablab4Context db = new())
            {
                var updatedLocalization = db.Localizations.Where(l => l.Id == id).FirstOrDefault();

                int kodPocztowyId;
                if (Int32.TryParse(changeLocalizationIdTextBox.Text, out _))
                {
                    kodPocztowyId = Int32.Parse(changeLocalizationKodPocztowyIdTextBox.Text);

                    var foreingKodPocztowy = db.KodyPocztowes.Where(kp => kp.Id == id).FirstOrDefault();

                    if (foreingKodPocztowy == null)
                    {

```

```
        MessageBox.Show($"You are trying to bind a new object to a non-existent 'Kode Pocztowy'");
    }
    else
    {
        updatedLocalization.KodyPocztoweId = kodPocztowyId;
        updatedLocalization.Name = changeLocalizationNameTextBox.Text;
        updatedLocalization.Description = changeLocalizationDescriptionText-
Box.Text;

        db.SaveChanges();
    }
    else
    {
        MessageBox.Show($"'Kod Pocztowy Id' must be Integer");
    }
}

clearKodyResultDataGridView();
clearLocalizationsResultDataGridView();
fillKodyDataGridView();
fillLocalizationsDataGridView();
}
catch (Exception ex)
{
    MessageBox.Show($"Occured error {ex.Message}");
}
finally
{
    changeLocalizationIdTextBox.Text = "";
    changeLocalizationDescriptionTextBox.Text = "";
    changeLocalizationNameTextBox.Text = "";
    changeLocalizationKodPocztowyIdTextBox.Text = "";
    localizationChangeGroupBox.Visible = false;
}
}
```

Snippet 9. Delete Localization Button

```
private void deleteLocalizationButton_Click(object sender, EventArgs e)
{
    try
    {
        using (Ajppablab4Context db = new())
        {
            int? id = null;
            if (Int32.TryParse(changeLocalizationIdTextBox.Text, out _)) id =
Int32.Parse(changeLocalizationIdTextBox.Text);

            var deletedLocalization = db.Localizations.Where(kp => kp.Id == id).FirstOrDefault();
            db.Localizations.Remove(deletedLocalization);
            db.SaveChanges();
        }

        clearKodyResultDataGridView();
        clearLocalizationsResultDataGridView();
        fillKodyDataGridView();
        fillLocalizationsDataGridView();
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Occured error {ex.Message}");
    }
    finally
    {
        changeLocalizationIdTextBox.Text = "";
        changeLocalizationDescriptionTextBox.Text = "";
        changeLocalizationNameTextBox.Text = "";
        changeLocalizationKodPocztowyIdTextBox.Text = "";
    }
}
```

```
        localizationChangeGroupBox.Visible = false;  
    }  
}
```

8. Wnioski

W trakcie wykonania laboratorium udało się jeszcze bardziej poszerzyć wiedzę zarówno jak praktyczną i teoretyczną w zakresie narzędzi ORM i aplikacji typu Windows Forms, a także przećwiczyć wymianę informacji pomiędzy kontrolkami a bazą danych.

9. Bibliografia

1. Źródła pomocnicze

- a. Dokumentacja Microsoft [<https://learn.microsoft.com/>], dostęp: 25.06.2023
- b. Kolumny w DataGridView [<https://wpf-tutorial.com/pl/89/kontrolka-datagrid/kolumny-w-datagrid/>], dostęp: 25.06.2023
- c. Guide to ADO.NET and working with databases in .NET 6 (oryg. Руководство по ADO.NET и работе с базами данных в .NET 6) [<https://metanit.com/sharp/adonetcore/>], dostęp: 25.06.2023
- d. Entity Framework Core Series [<https://code-maze.com/entity-framework-core-series/>], dostęp: 25.06.2023

10. Spis ilustracji

Rysunek 1. Schemat tabeli Kody_Pocztowe	5
Rysunek 2. Schemat tabeli Localization	5
Rysunek 3. Struktura projektu.....	6
Rysunek 4. Kontekst bazy danych	7
Rysunek 5. Widok aplikacji	8
Rysunek 6. Wyświetlanie powiązanych danych	8
Rysunek 7. Modyfikacja kodu pocztowego	9
Rysunek 8. Modyfikacja lokalizacji.....	10

11. Spis snippetów

Snippet 1. Skrypt do zainstalowania EF.....	6
---------------------------------------------	---

Snippet 2. Polecenie Scaffold-DbContext.....	6
Snippet 3. Search Button.....	10
Snippet 4. Edit Kod_Pocztowy Button	11
Snippet 5. Change Kod_Pocztowy Button.....	12
Snippet 6. Delet Kod_Pocztowy Button	13
Snippet 7. Edit Localization Button	13
Snippet 8. Changes Localization Button	14
Snippet 9. Delete Localization Button	15