**Лабораторна робота №5**

**Мета роботи:** набуття практичних навичок створення процедур, що зберігаються та їх виклик із зовнішніх додатків.

Хід роботи

**Реалізувати інтерфейс програми із використанням WPF**

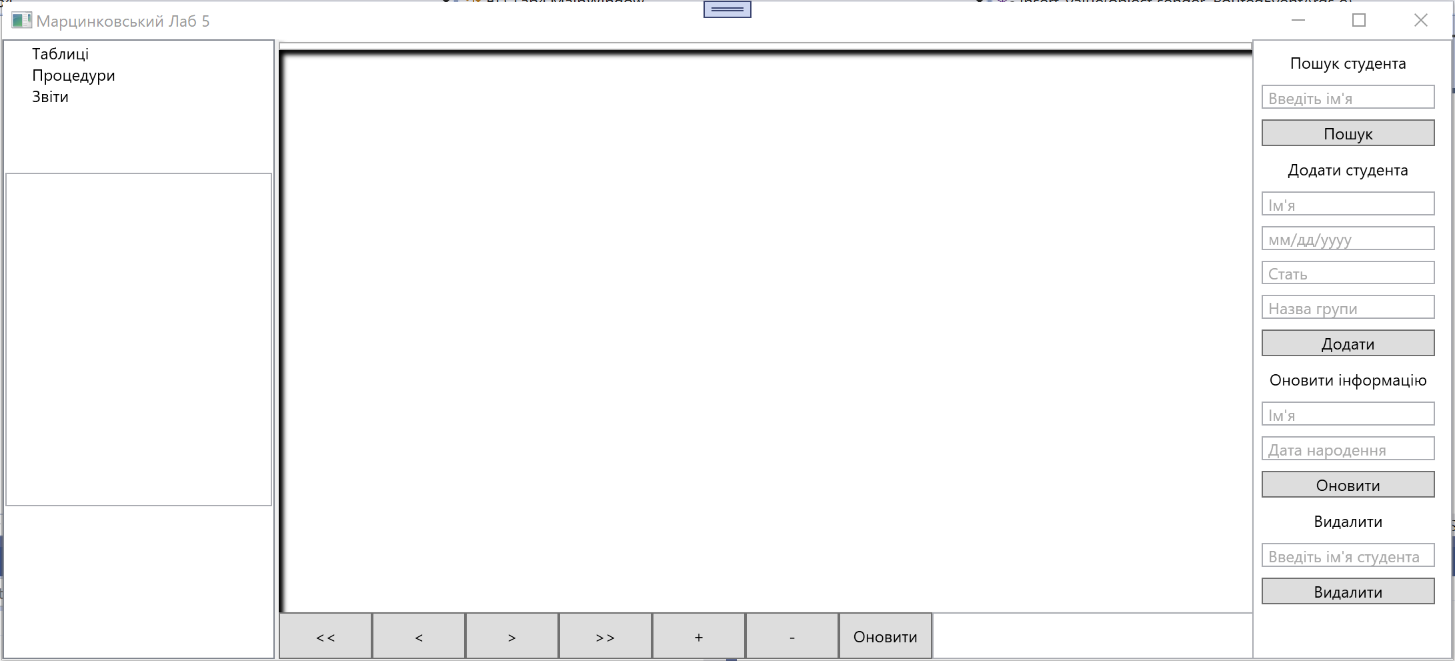


Рис. 5.1 Інтерфейс дадатку.

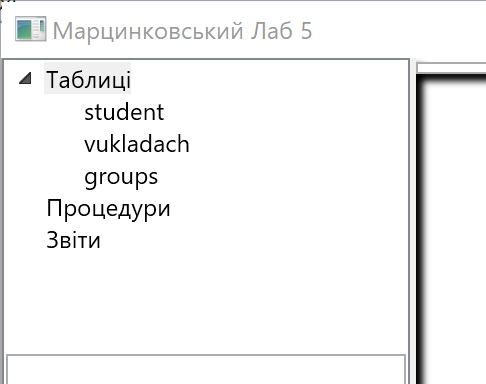


Рис. 5.2 Перегляд доступних таблиць.

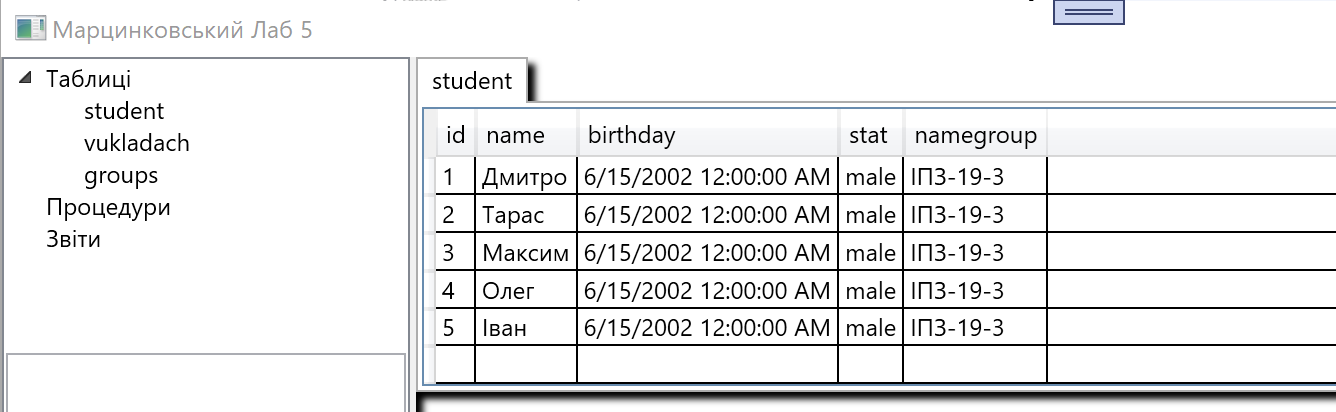


Рис. 5.3 Перегляд вмісту таблиці student.

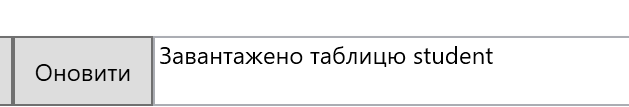


Рис. 5.4 Повідомлення для користувача.

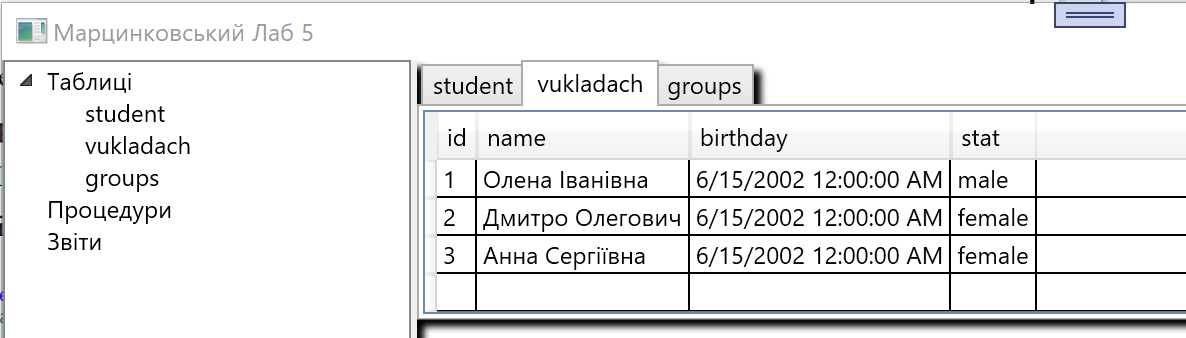


Рис. 5.5 Перегляд вмісту таблиці vukladach.

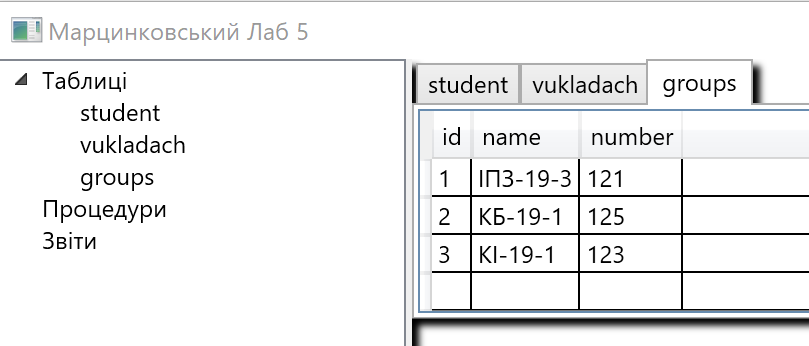


Рис. 5.6 Перегляд вмісту таблиці groups.

**Процедура пошуку даних:**

create proc Find\_student

@name varchar(20)

as

select [name], birthday, stat, name\_group

from student

where [name] like @name;

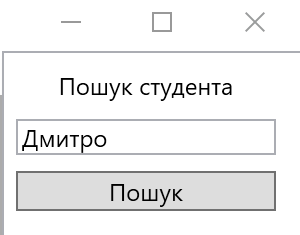


Рис. 5.7 Введення даних.

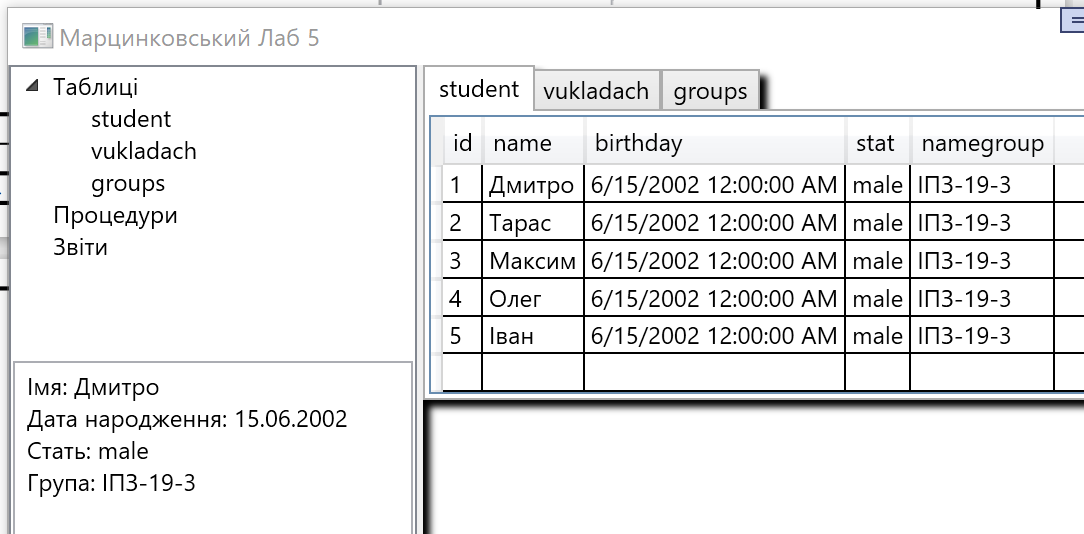


Рис. 5.8 Результат пошуку.

**Процедура вставки даних:**

create proc Add\_student

@name varchar(50),

@birthday varchar(20),

@stat varchar(20),

@name\_group varchar(10)

as

begin

insert into student

values(@name, @birthday, @stat, @name\_group)

end;

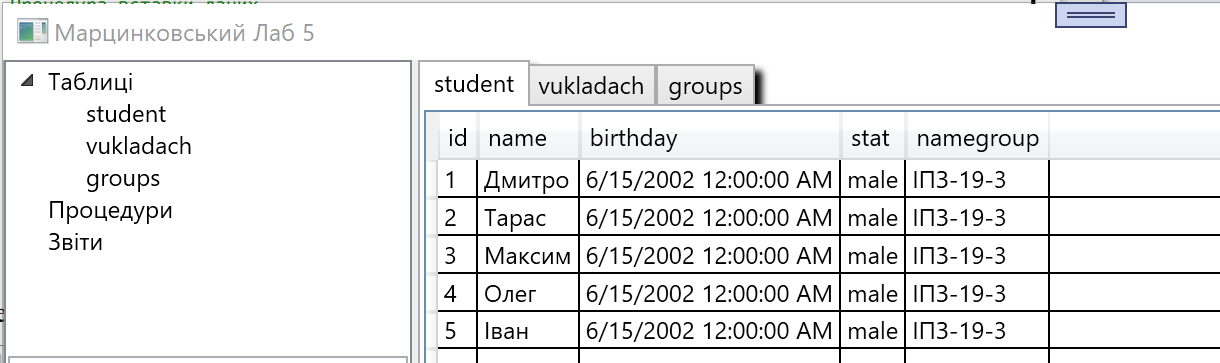


Рис. 5.9 Вміст таблиці до додавання.

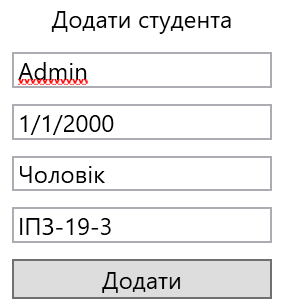


Рис. 5.10 Введення даних.

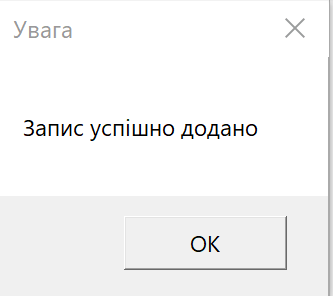


Рис. 5.11 Вивід повідомлення.

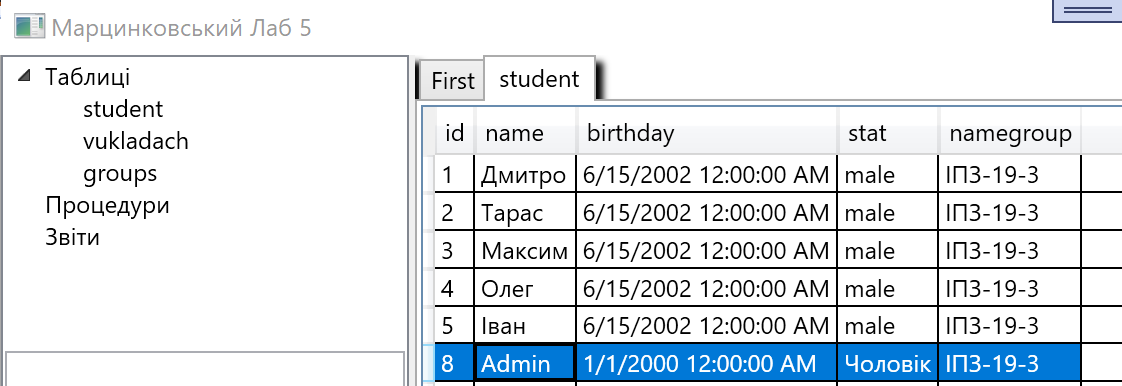


Рис. 5.12 Успішне додавання.

**Процедура оновлення даних:**

create proc Update\_student

@name varchar(50),

@birthday varchar(30)

as

begin

update student

set birthday=@birthday

where [name]=@name;

end;

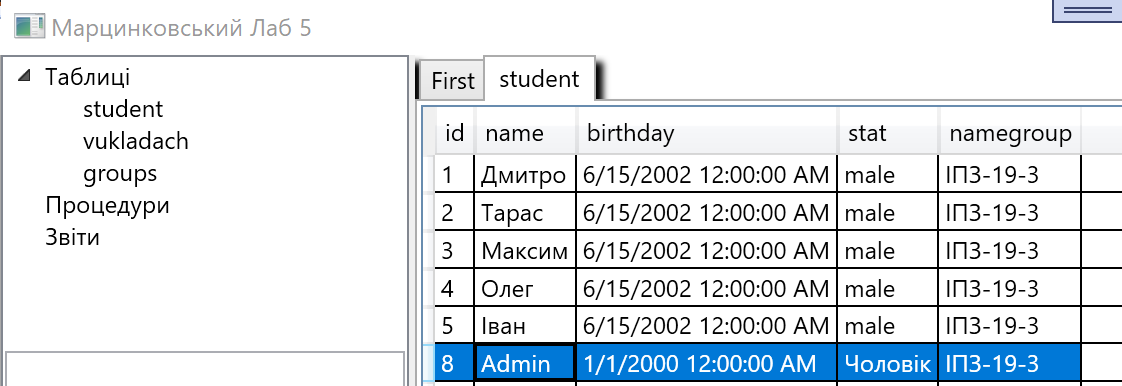


Рис. 5.13 Запис до оновлення.

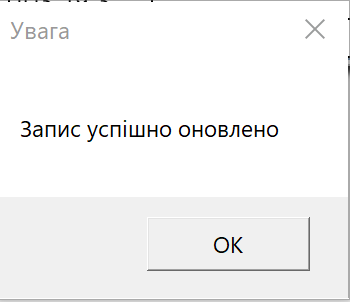


Рис. 5.14 Вивід повідомлення.

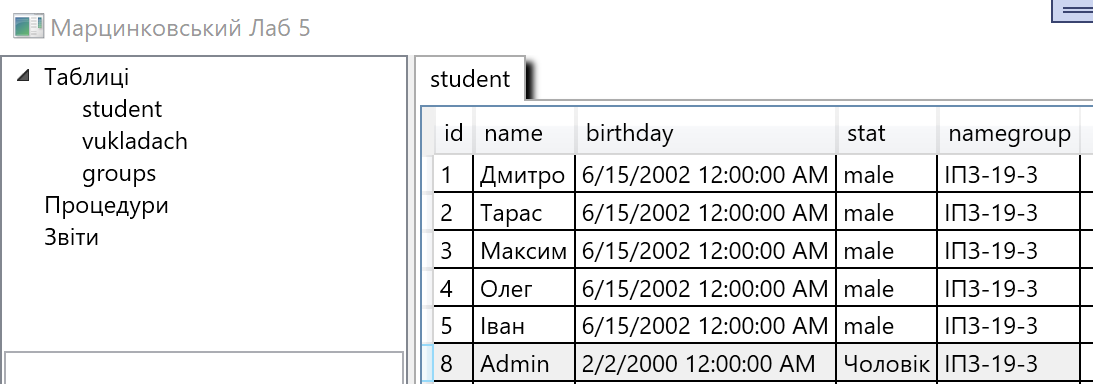


Рис. 5.15 Запис після оновлення

**Процедура видалення даних:**

create proc Delete\_student

@name varchar(50)

as

begin

delete

from student

where [name]=@name;

end;

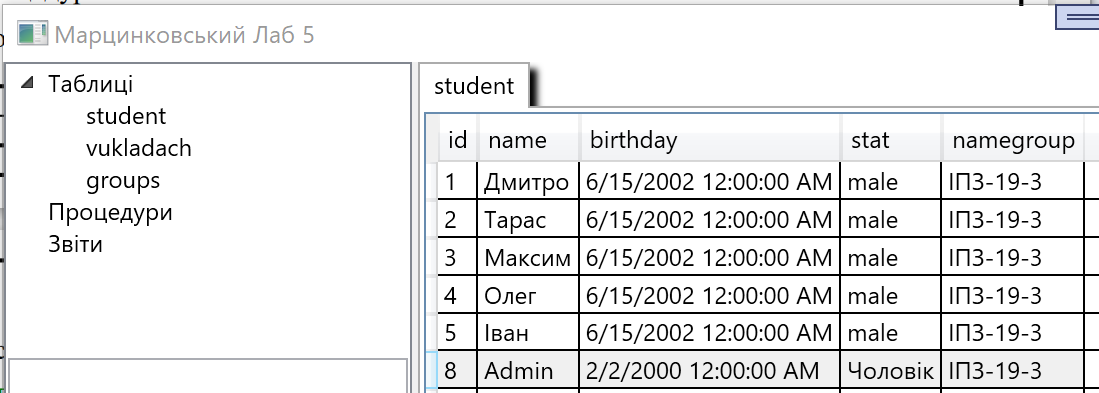


Рис. 5.16 Таблиця до видалення.

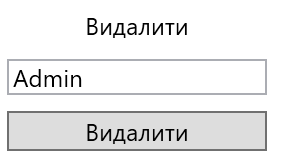


Рис. 5.17 Введення даних.

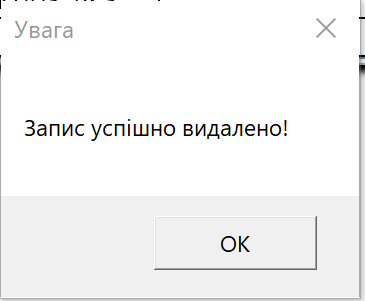


Рис. 5.18 Вивід повідомлення.

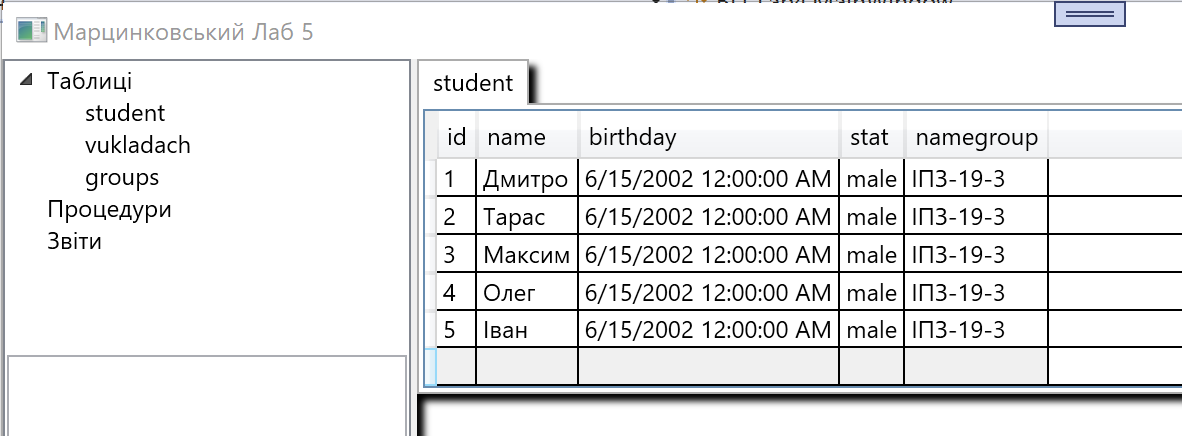


Рис. 5.19 Таблиця після видалення.

**Вміст файлу App.config:**

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<startup>

<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.7.2" />

</startup>

<connectionStrings>

<add name="Connect1" providerName="System.Data.SqlClient" connectionString="Data Source =DESKTOP-LO99G13; Initial Catalog =laboratory5; Integrated Security =True" />

</connectionStrings>

</configuration>

**Вміст файлу MainWindows.xalm.cs:**

using System;

using System.Windows;

using System.Windows.Controls;

using System.Data;

using System.Data.SqlClient;

using System.Configuration;

namespace Lab5

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

SqlConnection con;

DataSet ds;

public MainWindow()

{

InitializeComponent();

con = new SqlConnection(ConfigurationManager.ConnectionStrings["Connect1"].ConnectionString);

ds = new DataSet();

}

private void Table\_Selected(object sender, RoutedEventArgs e)

{

SqlCommand s = new SqlCommand();

TreeViewItem t = (TreeViewItem)sender;

t.Items.Clear();

if (t.Header.ToString() == "Таблиці")

{

s.CommandText = "SELECT TABLE\_NAME FROM INFORMATION\_SCHEMA.TABLES where Not(TABLE\_NAME like 'sysdiagrams')";

s.Connection = con;

}

if (t.Header.ToString() == "Процедури")

{

s.CommandText = "SELECT name FROM sys.objects o WHERE o.[Type] = 'P' and not(name like 'sp%')";

s.Connection = con;

}

con.Open();

SqlDataReader R = s.ExecuteReader();

if ((R.HasRows) & (t.Header.ToString() == "Таблиці"))

{

while (R.Read())

{

TreeViewItem ts = new TreeViewItem();

ts.Header = R[0].ToString();

ts.Selected += Table\_Selected1;

t.Items.Add(ts);

}

}

if ((R.HasRows) & (t.Header.ToString() == "Процедури"))

{

while (R.Read())

{

TreeViewItem ts = new TreeViewItem();

ts.Header = R[0].ToString();

ts.Selected += Proc\_Selected;

t.Items.Add(ts);

}

}

con.Close();

}

private void Table\_Selected1(object sender, RoutedEventArgs e)

{

TreeViewItem t = (TreeViewItem)sender;

SqlCommand s = new SqlCommand("SELECT \* from " + t.Header.ToString(), con);

con.Open();

SqlDataAdapter DA = new SqlDataAdapter(s);

DA.Fill(ds, t.Header.ToString());

DataGrid d = new DataGrid();

d.AutoGenerateColumns = true;

d.ItemsSource = ds.Tables[t.Header.ToString()].DefaultView;

ContextMenu c = new ContextMenu();

MenuItem m = new MenuItem();

m.Header = "Закрити";

m.Click += MenuItem\_Click;

c.Items.Add(m);

tabs.Items.Add(

new TabItem

{

Header = t.Header.ToString(),

Content = d,

ContextMenu = c

});

con.Close();

Status.Text = "Завантажено таблицю " + t.Header.ToString();

}

private void MenuItem\_Click(object sender, RoutedEventArgs e)

{

tabs.Items.RemoveAt(tabs.SelectedIndex);

}

private void Proc\_Selected(object sender, RoutedEventArgs e)

{

TreeViewItem t = (TreeViewItem)sender;

SqlCommand s = new SqlCommand();

s.CommandType = CommandType.StoredProcedure;

s.CommandText = (string)t.Header;

s.Connection = con;

con.Open();

SqlDataAdapter DA = new SqlDataAdapter(s);

DA.Fill(ds, (string)t.Header);

DataGrid d = new DataGrid();

d.AutoGenerateColumns = true;

d.ItemsSource = ds.Tables[(string)t.Header].DefaultView;

ContextMenu c = new ContextMenu();

MenuItem m = new MenuItem();

m.Header = "Закрити";

m.Click += MenuItem\_Click;

c.Items.Add(m);

tabs.Items.Add(

new TabItem

{

Header = t.Header.ToString(),

Content = d,

ContextMenu = c

});

con.Close();

Status.Text = "Завантажено таблицю " + t.Header.ToString();

}

private void Proc\_with\_param1(object sender, RoutedEventArgs e)

{

SqlCommand s = new SqlCommand("Find\_student", con);

SqlDataAdapter DA = new SqlDataAdapter(s);

s.CommandType = CommandType.StoredProcedure;

s.Parameters.Add("@name", SqlDbType.VarChar, 50);

s.Parameters["@name"].Value = Convert.ToString(Name\_text.Text);

Status.Text = Convert.ToString(s.Parameters["@name"].Value);

con.Open();

SqlDataReader reader = s.ExecuteReader();

while (reader.Read())

{

string name = reader.GetString(0);

DateTime birthday = reader.GetDateTime(1);

string stat = reader.GetString(2);

string group = reader.GetString(3);

rez.Items.Add("Імя: " + name + "\nДата народження: " + birthday.ToString("dd/MM/yyyy") + "\nСтать: " + stat + "\nГрупа: " + group);

}

reader.Close();

con.Close();

}

private void Insert\_value(object sender, RoutedEventArgs e)

{

con.Open();

SqlCommand s = new SqlCommand("Add\_student", con);

s.Transaction = con.BeginTransaction(System.Data.IsolationLevel.Serializable);

try

{

SqlDataAdapter DA = new SqlDataAdapter(s);

s.CommandType = CommandType.StoredProcedure;

s.Parameters.Add("@name", SqlDbType.VarChar, 50);

s.Parameters.Add("@birthday", SqlDbType.VarChar, 20);

s.Parameters.Add("@stat", SqlDbType.VarChar, 20);

s.Parameters.Add("@name\_group", SqlDbType.VarChar, 10);

s.Parameters["@name"].Value = Convert.ToString(name.Text);

s.Parameters["@birthday"].Value = Convert.ToString(birthday.Text);

s.Parameters["@stat"].Value = Convert.ToString(sex.Text);

s.Parameters["@name\_group"].Value = Convert.ToString(group.Text);

DA.InsertCommand = s;

SqlCommandBuilder comandbuilder = new SqlCommandBuilder(DA);

DA.TableMappings.Add("Table", "student");

DA.Update(ds);

DA.Fill(ds);

DataTable st = ds.Tables["student"];

ds.AcceptChanges();

s.ExecuteNonQuery();

s.Transaction.Commit();

MessageBox.Show("Запис успішно додано", "Увага", MessageBoxButton.OK);

}

catch (Exception ex)

{

s.Transaction.Rollback();

MessageBox.Show("Помилка: " + ex.Message, "Увага", MessageBoxButton.OK);

}

finally

{

con.Close();

}

}

private void Update\_value(object sender, RoutedEventArgs e)

{

con.Open();

SqlCommand s = new SqlCommand("Update\_student", con);

s.Transaction = con.BeginTransaction(System.Data.IsolationLevel.Serializable);

try

{

SqlDataAdapter DA = new SqlDataAdapter(s);

s.CommandType = CommandType.StoredProcedure;

s.Parameters.Add("@name", SqlDbType.VarChar, 50);

s.Parameters.Add("@birthday", SqlDbType.VarChar, 30);

s.Parameters["@name"].Value = Convert.ToString(name\_update.Text);

s.Parameters["@birthday"].Value = Convert.ToString(birthday\_update.Text);

DA.UpdateCommand = s;

DA.TableMappings.Add("Table", "student");

DA.Update(ds);

DA.Fill(ds);

ds.AcceptChanges();

s.ExecuteNonQuery();

s.Transaction.Commit();

MessageBox.Show("Запис успішно оновлено", "Увага", MessageBoxButton.OK);

}

catch (Exception ex)

{

s.Transaction.Rollback();

MessageBox.Show("Помилка: " + ex.Message, "Увага", MessageBoxButton.OK);

}

finally

{

con.Close();

}

}

private void Delete\_value(object sender, RoutedEventArgs e)

{

con.Open();

SqlCommand s = new SqlCommand("Delete\_student", con);

s.Transaction = con.BeginTransaction(System.Data.IsolationLevel.Serializable);

try

{

SqlDataAdapter DA = new SqlDataAdapter(s);

s.CommandType = CommandType.StoredProcedure;

s.Parameters.Add("@name", SqlDbType.VarChar, 50);

s.Parameters["@name"].Value = Convert.ToString(del\_name.Text);

DA.UpdateCommand = s;

DA.TableMappings.Add("Table", "student");

DA.Update(ds);

DA.Fill(ds);

ds.AcceptChanges();

s.ExecuteNonQuery();

s.Transaction.Commit();

MessageBox.Show("Запис успішно видалено!", "Увага", MessageBoxButton.OK);

}

catch (Exception ex)

{

s.Transaction.Rollback();

MessageBox.Show("Помилка: " + ex.Message, "Увага", MessageBoxButton.OK);

}

finally

{

con.Close();

}

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

}

}

}

**XAML:**

<Window x:Class="BD\_Lab4.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:BD\_Lab4"

mc:Ignorable="d"

Title="Марцинковський Лаб 5

" Height="500" Width="1100" WindowStartupLocation="CenterScreen">

<!--Title="MainWindow" Height="500" Width="1100" ResizeMode="NoResize" WindowStartupLocation="CenterScreen">-->

<Grid>

<Grid.Effect>

<DropShadowEffect/>

</Grid.Effect>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="Auto" />

<ColumnDefinition Width="3" />

<ColumnDefinition Width="14\*" />

<ColumnDefinition Width="333\*" />

<ColumnDefinition Width="150" />

</Grid.ColumnDefinitions>

<GridSplitter Grid.Column="1" x:Name="gridSplitter" HorizontalAlignment="Center" Width="3" VerticalAlignment="Stretch" />

<TreeView Grid.Column="0" x:Name="treeView" HorizontalAlignment="Left" VerticalAlignment="Stretch" MinWidth="120">

<TreeViewItem x:Name="Table" Header="Таблиці" Selected="Table\_Selected" />

<TreeViewItem x:Name="Proc" Header="Процедури" Selected="Table\_Selected" />

<TreeViewItem x:Name="Report" Header="Звіти" />

<ListView x:Name="rez" Width="200" MinHeight="250" Height="Auto" Margin="-20,50,0,0"/>

</TreeView>

<TabControl Grid.Column="2" Grid.Row ="0" x:Name="tabs" HorizontalAlignment="Stretch" VerticalAlignment="Top" Height="Auto" Grid.ColumnSpan="2" >

<TabItem Header="First">

<TabItem.ContextMenu>

<ContextMenu>

<MenuItem Header="Закрыть" Click="MenuItem\_Click" />

</ContextMenu>

</TabItem.ContextMenu>

<DataGrid x:Name="DG" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ></DataGrid>

</TabItem>

</TabControl>

<DockPanel LastChildFill="False" Grid.Column="2" HorizontalAlignment="Stretch" Height="35" VerticalAlignment="Bottom" Grid.ColumnSpan="2">

<Button Content="&lt;&lt;" Width="70" />

<Button Content="&lt;" Width="70" />

<Button Content=">" Width="70" />

<Button Content=">>" Width="70" />

<Button Content="+" Width="70" />

<Button Content="-" Width="70" />

<Button Content="Оновити" Width="70" Click="Button\_Click" />

<TextBox Width="Auto" MinWidth="270" x:Name="Status" DockPanel.Dock="Right"/>

</DockPanel>

<ListView x:Name="listView1" MinWidth="120" Grid.Column="4" >

<Label x:Name="Table\_name" Content="Пошук студента" HorizontalContentAlignment="Center" Width="130"/>

<TextBox x:Name="Name\_text" SpellCheck.IsEnabled="True" Width="130" />

<TextBlock IsHitTestVisible="False" Text="Введіть ім'я" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=Name\_text}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<Button x:Name="Button1" Content="Пошук" Click="Proc\_with\_param1" Width="130" />

<Label x:Name="Insert" Content="Додати студента" HorizontalContentAlignment="Center" Width="130"/>

<TextBox x:Name="name" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="Ім'я" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=name}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<TextBox x:Name="birthday" Text="" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="мм/дд/yyyy" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=birthday}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<TextBox x:Name="sex" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="Стать" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=sex}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<TextBox x:Name="group" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="Назва групи" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=group}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<Button x:Name="Button2" Content="Додати" Click="Insert\_value" Width="130" />

<Label x:Name="Update" Content="Оновити інформацію" HorizontalContentAlignment="Center" Width="130"/>

<TextBox x:Name="name\_update" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="Ім'я" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=name\_update}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<TextBox x:Name="birthday\_update" SpellCheck.IsEnabled="True" Width="130"/>

<TextBlock IsHitTestVisible="False" Text="Дата народення" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=birthday\_update}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<Button x:Name="Button\_update" Content="Оновити" Click="Update\_value" Width="130" />

<Label Content="Видалити" HorizontalContentAlignment="Center" Width="130" />

<TextBox x:Name="del\_name" SpellCheck.IsEnabled="True" Width="130" />

<TextBlock IsHitTestVisible="False" Text="Введіть ім'я студента" Padding="5,0" Height="16" Width="130" Margin="0,-25,0,0" Foreground="DarkGray">

<TextBlock.Style>

<Style TargetType="{x:Type TextBlock}">

<Setter Property="Visibility" Value="Collapsed"/>

<Style.Triggers>

<DataTrigger Binding="{Binding Text, ElementName=del\_name}" Value="">

<Setter Property="Visibility" Value="Visible"/>

</DataTrigger>

</Style.Triggers>

</Style>

</TextBlock.Style>

</TextBlock>

<Button x:Name="Del\_button" Content="Видалити" Click="Delete\_value" Width="130" />

</ListView>

</Grid>

</Window>

**Висновок**: в ході виконання лабороторної роботи, отримав практичні навички створення процедур, що зберігаються та їх виклик із зовнішніх додатків..