Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский экономический университет им. Г.В. Плеханова»

Московский приборостроительный техникум

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

Общепрофессиональная дисциплина: ОП 08 Основы проектирования баз данных

МПТ.09.02.07-П.ОП.08.ОПБД.П50-7-20.10.22

Отчёт к практической работе № 7

«Реализация манипулирования данными и выборки данных из базы данных»

Тема: «Разработка базы данных на примере предметной области: ««Ресторан». Реализация

подсистем: учёт и обработка заказов, учёт и реализация меню».

|  |  |
| --- | --- |
| Проверил: | Выполнил: |
| Щаников И.М.\_\_\_\_\_\_\_\_\_\_ | Студент группы |
| (\_\_\_\_\_\_\_\_)\_\_\_\_\_\_\_\_\_\_\_\_\_ | П50-7-20 |
| «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022г. | \_\_\_\_\_\_\_\_\_\_\_\_Огурцов А.А. |
|  | «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022г. |

2022

СОДЕРЖАНИЕ

[1. ЦЕЛИ ПРАКТИЧЕСКОЙ РАБОТЫ 3](#_Toc92804752)

[2. ЭТАПЫ ВЫПОЛНЕНИЯ 4](#_Toc92804753)

[3. ВЫВОД 5](#_Toc92804754)

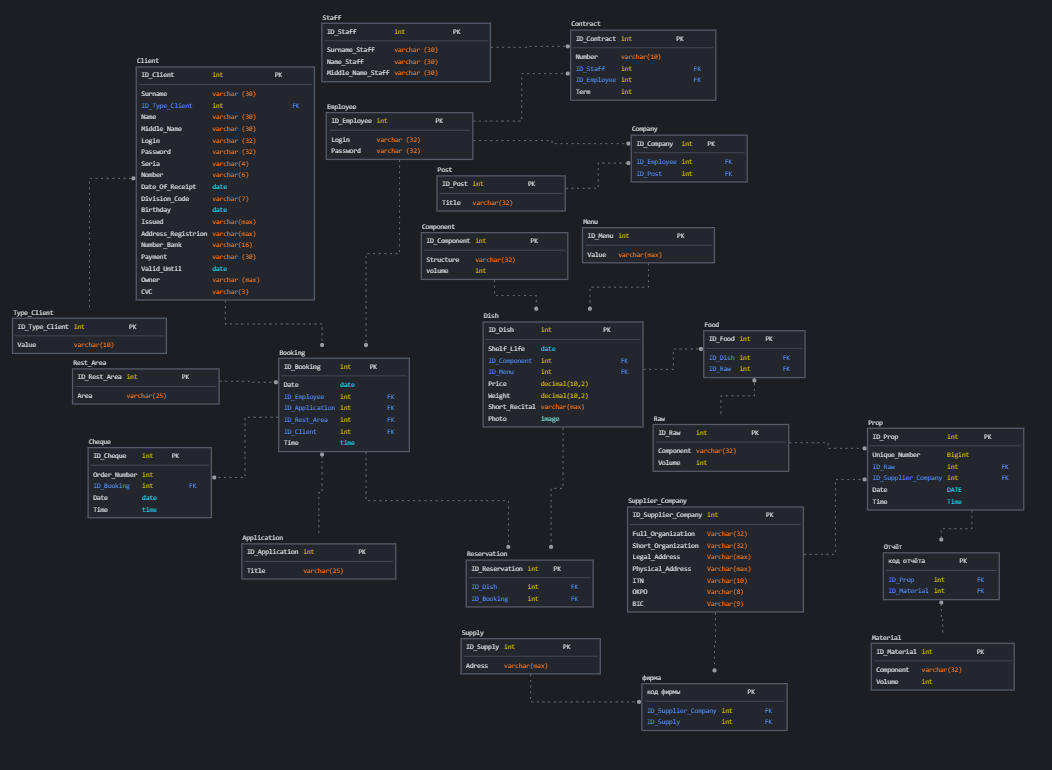
[4. СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ 6](#_Toc92804755)

1. ЦЕЛИ ПРАКТИЧЕСКОЙ РАБОТЫ

На основании разработанной структуры базы данных с помощью языка DDL, произвести заполнение структуры базы данных и выборку данных из таблиц базы данных.

ЭТАПЫ ВЫПОЛНЕНИЯ

1. На даталогической модели данных отметить те таблицы, которые относятся к одной из подсистем и не относятся к операционным таблицам (Приложение пишется для администратора системы, в задачу которого входит актуализация данных в справочных таблицах);



1. В инструментальной среде реализовать универсальный класс для работы с запросами к базе данных;

using System.Data;

using System.Data.SqlClient;

namespace WPF\_BD\_7

{

class DB\_Class

{

//public static string ConnStr = " Data Source= ACER-AN\\SQLXARD;Initial Catalog=Restoran\_DB; Persist Security Info = true; User ID = sa; Password = 123";

public static string ConnStr = "Data Source= DESKTOP-8IQR0EI\\SQLEXPRESS;Initial Catalog=Restoran\_DB; Persist Security Info = true; User ID = {0}; Password = {1}";

public static string Users\_ID = "null", Password = "null", App\_Name = "Администратор - Продажа товара";

public SqlConnection conn = new SqlConnection(ConnStr);

public SqlCommand command = new SqlCommand();

public DataTable ResTable = new DataTable();

public SqlDependency dep = new SqlDependency();

public enum act { select, manipulation };

public void SqlExecute(string querty, act act)

{

command.Connection = conn;

command.CommandText = querty;

command.Notification = null;

switch (act)

{

case act.select:

dep.AddCommandDependency(command);

SqlDependency.Start(conn.ConnectionString);

conn.Open();

ResTable.Load(command.ExecuteReader());

conn.Close();

break;

case act.manipulation:

conn.Open();

command.ExecuteNonQuery();

conn.Close();

break;

}

}

}

}

1. Реализовать интерфейс для справочных таблиц;

|  |
| --- |
| Код разметки XAML – Администратор |
| <Window x:Class="WPF\_BD\_7.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:WPF\_BD\_7"  xmlns:ms ="http://schemas.xceed.com/wpf/xaml/toolkit"  mc:Ignorable="d"  Title="MainWindow" Height="450" Width="821">  <Grid>    <Grid.ColumnDefinitions>  <ColumnDefinition Width="700"/>  <ColumnDefinition Width="\*"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  </Grid.RowDefinitions>  <Button x:Name="gr1" Content="Поставка" Grid.Column="1" Grid.Row="1" Click="Button\_Click\_14" IsEnabled="False"/>  <Button x:Name="gr2" Content="Меню" Grid.Column="1" Grid.Row="2" Click="Button\_Click"/>  <Button x:Name="gr3" Content="Компонент" Grid.Column="1" Grid.Row="3" Click="Button\_Click\_1"/>  <Button x:Name="gr4" Content="Поставщик" Grid.Column="1" Grid.Row="4" Click="Button\_Click\_2"/>  <Grid x:Name="\_G1" Grid.RowSpan="6" Background="LightSlateGray" Visibility="Visible">  <Grid.ColumnDefinitions>  <ColumnDefinition Width="400"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  </Grid.RowDefinitions>  <DataGrid x:Name="\_dg\_supply" Grid.ColumnSpan="6" Grid.RowSpan="6" ColumnWidth="699"/>  <TextBox x:Name="\_supply" Grid.Row="6"/>  <Button Content="add\_supply" Grid.Column="1" Grid.Row="6" Click="Button\_Click\_3"/>  <Button Content="update\_supply" Grid.Column="2" Grid.Row="6" Click="Button\_Click\_4"/>  <Button Content="delete\_supply" Grid.Column="3" Grid.Row="6" Click="Button\_Click\_5"/>  </Grid>  <Grid x:Name="\_G2" Grid.RowSpan="6" Background="LightSlateGray" Visibility="Hidden">  <Grid.ColumnDefinitions>  <ColumnDefinition Width="400"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  </Grid.RowDefinitions>  <DataGrid x:Name="\_dg\_menu" Grid.ColumnSpan="6" Grid.RowSpan="6" ColumnWidth="699"/>  <TextBox x:Name="\_menu" Grid.Row="6"/>  <Button Content="add\_menu" Grid.Column="1" Grid.Row="6" Click="Button\_Click\_6"/>  <Button Content="update\_menu" Grid.Column="2" Grid.Row="6" Click="Button\_Click\_7"/>  <Button Content="delete\_menu" Grid.Column="3" Grid.Row="6" Click="Button\_Click\_8"/>  </Grid>  <Grid x:Name="\_G3" Grid.RowSpan="6" Background="LightSlateGray" Visibility="Hidden">  <Grid.ColumnDefinitions>  <ColumnDefinition Width="300"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  </Grid.RowDefinitions>  <DataGrid x:Name="\_dg\_component" Grid.ColumnSpan="6" Grid.RowSpan="6" ColumnWidth="345"/>  <TextBox x:Name="\_component" Grid.Row="6"/>  <ms:IntegerUpDown x:Name="\_int\_component" Grid.Column="1" Grid.Row="6" FontSize="22" />  <Button Content="add\_component" Grid.Column="2" Grid.Row="6" Click="Button\_Click\_9"/>  <Button Content="update\_component" Grid.Column="3" Grid.Row="6" Click="Button\_Click\_11"/>  <Button Content="delete\_component" Grid.Column="4" Grid.Row="6" Click="Button\_Click\_10"/>  </Grid>  <Grid x:Name="\_G4" Grid.RowSpan="6" Background="LightSlateGray" Visibility="Hidden">  <Grid.ColumnDefinitions>  <ColumnDefinition Width="140"/>  <ColumnDefinition Width="80"/>  <ColumnDefinition Width="80"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  <ColumnDefinition Width="100"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="60"/>  <RowDefinition Height="40"/>  <RowDefinition Height="40"/>  <RowDefinition Height="40"/>  <RowDefinition Height="40"/>  <RowDefinition Height="40"/>  <RowDefinition Height="40"/>  <RowDefinition Height="60"/>  <RowDefinition Height="60"/>  </Grid.RowDefinitions>  <TextBlock Text="Полное название организации" Grid.Row="2" TextWrapping="Wrap"/>  <TextBlock Text="Сокращённое название организации" Grid.Row="3" TextWrapping="Wrap"/>  <TextBlock Text="ИНН" Grid.Row="4"/>  <TextBlock Text="ОКПО" Grid.Row="5"/>  <TextBlock Text="БИК" Grid.Row="6"/>  <TextBlock Text="Физический адрес" Grid.Row="7"/>  <TextBlock Text="Юредический адрес" Grid.Row="8"/>  <TextBox x:Name="\_sp\_All" Grid.Column="1" Grid.ColumnSpan="2" Grid.Row="2"/>  <TextBox x:Name="\_sp\_Short" Grid.Column="1" Grid.ColumnSpan="2" Grid.Row="3"/>  <TextBox x:Name="\_sp\_ITN" Grid.Column="1" Grid.ColumnSpan="2" Grid.Row="4"/>  <TextBox x:Name="\_sp\_OKPO" Grid.Column="1" Grid.ColumnSpan="2" Grid.Row="5"/>  <TextBox x:Name="\_sp\_BIC" Grid.Column="1" Grid.ColumnSpan="2" Grid.Row="6"/>  <TextBox x:Name="\_sp\_Fis\_" Grid.Column="1" Grid.ColumnSpan="3" Grid.Row="7"/>  <TextBox x:Name="\_sp\_Sec" Grid.Column="1" Grid.ColumnSpan="3" Grid.Row="8"/>  <DataGrid x:Name="\_dg\_supplier\_Company" Grid.Column="3" Grid.ColumnSpan="4" Grid.RowSpan="7"/>  <Button Content="add\_men" Grid.Column="4" Grid.Row="8" Click="Button\_Click\_12"/>  <Button Content="update\_men" Grid.Column="5" Grid.Row="8" Click="Button\_Click\_15"/>  <Button Content="delete\_men" Grid.Column="6" Grid.Row="8" Click="Button\_Click\_13"/>  </Grid>  </Grid>  </Window> |
| Результат |
| Вкладка «Поставка»    Вкладка «Меню»    Вкладка «Компонент»    Вкладка «Поставщик» |

1. Реализовать функции добавления, изменения и удаления данных;

|  |  |
| --- | --- |
| Иллюстрация справочника | Вывод данных в справочник(и) |
|  | private void PostavkaFill()  {  if(\_dg\_supply != null)  {  Action action = () =>  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("select [ID\_Supply], [Adress\_Supply] from [dbo].[Supply]", DB\_Class.act.select);  db\_cl.dep.OnChange += Dep\_OnChange\_postavka;  \_dg\_supply.ItemsSource = db\_cl.ResTable.DefaultView;  \_dg\_supply.Columns[0].Visibility = Visibility.Hidden;  \_dg\_supply.Columns[1].Header = "Адрес";  };  Dispatcher.Invoke(action);  }  }  private void Dep\_OnChange\_postavka(object sender, SqlNotificationEventArgs e)  {  if (e.Info != SqlNotificationInfo.Invalid)  PostavkaFill();  } |
| Добавление данных |
| private void Button\_Click\_3(object sender, RoutedEventArgs e)  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("insert into [dbo].[Supply] ([Adress\_Supply]) values('"+\_supply.Text+"')", DB\_Class.act.manipulation);  \_supply.Clear();  } |
| Изменение данных |
| private void Button\_Click\_4(object sender, RoutedEventArgs e)  {  if (\_dg\_supply.Items.Count != 0 & \_dg\_supply.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_supply.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("update [dbo].[Supply] set " +  "[Adress\_Supply] = '"+\_supply.Text+"'"+  "where [ID\_Supply] = " + dataRowView[0], DB\_Class.act.manipulation);  }  } |
| Удаление данных |
| private void Button\_Click\_5(object sender, RoutedEventArgs e)  {  switch (MessageBox.Show("Удалит выбранную запись?", DB\_Class.App\_Name, MessageBoxButton.YesNo, MessageBoxImage.Question))  {  case MessageBoxResult.Yes:  if (\_dg\_supply.Items.Count != 0 & \_dg\_supply.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_supply.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("delete from [dbo].[Supply] where [ID\_Supply] = " + dataRowView[0], DB\_Class.act.manipulation);  }  break;  }  } |
| Иллюстрация справочника | Вывод данных в справочник(и) |
|  | private void MenuFill()  {  if (\_dg\_menu != null)  {  Action action = () =>  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("select [ID\_Menu], [Value\_Menu] from [dbo].[Menu]", DB\_Class.act.select);  db\_cl.dep.OnChange += Dep\_OnChange\_menu;  \_dg\_menu.ItemsSource = db\_cl.ResTable.DefaultView;  \_dg\_menu.Columns[0].Visibility = Visibility.Hidden;  \_dg\_menu.Columns[1].Header = "Название блюд меню";  };  Dispatcher.Invoke(action);  }  }  private void Dep\_OnChange\_menu(object sender, SqlNotificationEventArgs e)  {  if (e.Info != SqlNotificationInfo.Invalid)  MenuFill();  } |
| Добавление данных |
| private void Button\_Click\_6(object sender, RoutedEventArgs e)  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("insert into [dbo].[Menu] ([Value\_Menu]) values('" + \_menu.Text + "')", DB\_Class.act.manipulation);  \_menu.Clear();  } |
| Изменение данных |
| private void Button\_Click\_7(object sender, RoutedEventArgs e)  {  if (\_dg\_menu.Items.Count != 0 & \_dg\_menu.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_menu.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("update [dbo].[Menu] set " +  "[Value\_Menu] = '" + \_menu.Text + "'" +  "where [ID\_Menu] = " + dataRowView[0], DB\_Class.act.manipulation);  }  } |
| Удаление данных |
| private void Button\_Click\_8(object sender, RoutedEventArgs e)  {  switch (MessageBox.Show("Удалит выбранную запись?", DB\_Class.App\_Name, MessageBoxButton.YesNo, MessageBoxImage.Question))  {  case MessageBoxResult.Yes:  if (\_dg\_menu.Items.Count != 0 & \_dg\_menu.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_menu.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("delete from [dbo].[Menu] where [ID\_Menu] = " + dataRowView[0], DB\_Class.act.manipulation);  }  break;  }  } |
| Иллюстрация справочника | Вывод данных в справочник(и) |
|  | private void ComponentFill()  {  if (\_dg\_component != null)  {  Action action = () =>  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("select [ID\_Component], [Structure\_Component], [Volume\_Component] from [dbo].[Component]", DB\_Class.act.select);  db\_cl.dep.OnChange += Dep\_OnChange\_component;  \_dg\_component.ItemsSource = db\_cl.ResTable.DefaultView;  \_dg\_component.Columns[0].Visibility = Visibility.Hidden;  \_dg\_component.Columns[1].Header = "Название";  \_dg\_component.Columns[2].Header = "Количество";  };  Dispatcher.Invoke(action);  }  }  private void Dep\_OnChange\_component(object sender, SqlNotificationEventArgs e)  {  if (e.Info != SqlNotificationInfo.Invalid)  ComponentFill();  } |
| Добавление данных |
| private void Button\_Click\_9(object sender, RoutedEventArgs e)  {  if (\_int\_component.Value > -1)  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("insert into [dbo].[Component] ([Structure\_Component], [Volume\_Component]) values('" + \_component.Text + "'," + \_int\_component.Value + ")", DB\_Class.act.manipulation);  \_component.Clear();  }  } |
| Изменение данных |
| private void Button\_Click\_11(object sender, RoutedEventArgs e)  {  if (\_dg\_component.Items.Count != 0 & \_dg\_component.SelectedItems.Count != 0 & \_int\_component.Value > -1)  {  try  {  DataRowView dataRowView = (DataRowView)\_dg\_component.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("update [dbo].[Component] set " +  "[Structure\_Component] = '" + \_component.Text + "'" +  ",[Volume\_Component] = '" + \_int\_component.Value + "'" +  " where [ID\_Component] = " + dataRowView[0], DB\_Class.act.manipulation);  }  catch  {  MessageBox.Show("Неправильные значения");  }  }  } |
| Удаление данных |
| private void Button\_Click\_10(object sender, RoutedEventArgs e)  {  switch (MessageBox.Show("Удалит выбранную запись?", DB\_Class.App\_Name, MessageBoxButton.YesNo, MessageBoxImage.Question))  {  case MessageBoxResult.Yes:  if (\_dg\_component.Items.Count != 0 & \_dg\_component.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_component.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("delete from [dbo].[Component] where [ID\_Component] = " + dataRowView[0], DB\_Class.act.manipulation);  }  break;  }  } |
| Иллюстрация справочника | Вывод данных в справочник(и) |
|  | private void Supplier\_CompanyFill()  {  if (\_dg\_supplier\_Company != null)  {  Action action = () =>  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("select [ID\_Supplier\_Company], [Full\_Organization], [Short\_Organization], [Legal\_Address], [Physical\_Address], [ITN], [OKPO], [BIC] from [dbo].[Supplier\_Company]", DB\_Class.act.select);  db\_cl.dep.OnChange += Dep\_OnChange\_supplier\_company;  \_dg\_supplier\_Company.ItemsSource = db\_cl.ResTable.DefaultView;  \_dg\_supplier\_Company.Columns[0].Visibility = Visibility.Hidden;  \_dg\_supplier\_Company.Columns[1].Header = "Полное название организации";  \_dg\_supplier\_Company.Columns[2].Header = "Сокращённое название организации";  \_dg\_supplier\_Company.Columns[3].Header = "Физический адрес";  \_dg\_supplier\_Company.Columns[4].Header = "Юридический дрес";  \_dg\_supplier\_Company.Columns[5].Header = "ИНН";  \_dg\_supplier\_Company.Columns[6].Header = "ОКПО";  \_dg\_supplier\_Company.Columns[7].Header = "БИК";  };  Dispatcher.Invoke(action);  }  }  private void Dep\_OnChange\_supplier\_company(object sender, SqlNotificationEventArgs e)  {  if (e.Info != SqlNotificationInfo.Invalid)  Supplier\_CompanyFill();  } |
| Добавление данных |
| private void Button\_Click\_12(object sender, RoutedEventArgs e)  {  DB\_Class db\_cl = new DB\_Class();  db\_cl.SqlExecute("insert into [dbo].[Supplier\_Company] ([Full\_Organization], [Short\_Organization], [Legal\_Address], [Physical\_Address], [ITN], [OKPO], [BIC]) " +  "values('" + \_sp\_All.Text + "','" + \_sp\_Short.Text +  "','" + \_sp\_Fis\_.Text + "','" + \_sp\_Sec.Text +  "','" + \_sp\_ITN.Text + "','" + \_sp\_OKPO.Text +  "','" + \_sp\_BIC.Text + "')", DB\_Class.act.manipulation);  \_sp\_All.Clear();  \_sp\_Sec.Clear();  \_sp\_Short.Clear();  \_sp\_Fis\_.Clear();  \_sp\_OKPO.Clear();  \_sp\_ITN.Clear();  \_sp\_BIC.Clear();  } |
| Изменение данных |
| private void Button\_Click\_15(object sender, RoutedEventArgs e)  {  if (\_dg\_supplier\_Company.Items.Count != 0 & \_dg\_supplier\_Company.SelectedItems.Count != 0)  {  try  {  DataRowView dataRowView = (DataRowView)\_dg\_supplier\_Company.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("update [dbo].[Supplier\_Company] set " +  "[Full\_Organization] ='" + \_sp\_All.Text +  "', [Short\_Organization] ='" + \_sp\_Short.Text +  "', [Legal\_Address] ='" + \_sp\_Fis\_.Text +  "', [Physical\_Address] ='" + \_sp\_Sec.Text +  "', [ITN] ='" + \_sp\_ITN.Text +  "', [OKPO] ='" + \_sp\_OKPO.Text +  "', [BIC] ='" + \_sp\_BIC.Text + "'" +  "where [ID\_Supplier\_Company] = " + dataRowView[0], DB\_Class.act.manipulation);  }  catch  {  MessageBox.Show("Неправильные значения");  }  }  } |
| Удаление данных |
| private void Button\_Click\_13(object sender, RoutedEventArgs e)  {  switch (MessageBox.Show("Удалит выбранную запись?", DB\_Class.App\_Name, MessageBoxButton.YesNo, MessageBoxImage.Question))  {  case MessageBoxResult.Yes:  if (\_dg\_supplier\_Company.Items.Count != 0 & \_dg\_supplier\_Company.SelectedItems.Count != 0)  {  DataRowView dataRowView = (DataRowView)\_dg\_supplier\_Company.SelectedItems[0];  DB\_Class db\_class = new DB\_Class();  db\_class.SqlExecute("delete from [dbo].[Supplier\_Company] where [ID\_Supplier\_Company] = " + dataRowView[0], DB\_Class.act.manipulation);  }  break;  }  } |

1. Реализовать окно авторизации администратора, при этом администратор работает из-под логина и пароля от СУБД Microsoft SQL Server;

|  |
| --- |
| Код разметки XAML |
| <Window x:Class="WPF\_BD\_7.Window1"  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:WPF\_BD\_7"  mc:Ignorable="d"  Title="Покдлючение" Height="350" Width="800">  <Grid>  <Grid.ColumnDefinitions>  <ColumnDefinition Width="\*"/>  <ColumnDefinition Width="\*"/>  <ColumnDefinition Width="50"/>  <ColumnDefinition Width="\*"/>  <ColumnDefinition Width="\*"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  <RowDefinition Height="\*"/>  </Grid.RowDefinitions>  <Label Content="Подключение" Grid.ColumnSpan="5" FontSize="50" HorizontalContentAlignment="Center" VerticalContentAlignment="Center"/>  <TextBox x:Name="\_LOG" Grid.Row="1" Grid.Column="3" Grid.ColumnSpan="2" FontSize="22" VerticalContentAlignment="Center" HorizontalContentAlignment="Center"/>  <PasswordBox x:Name="\_PAS" Grid.Row="2" Grid.Column="3" Grid.ColumnSpan="2" FontSize="22" VerticalContentAlignment="Center" HorizontalContentAlignment="Center"/>  <TextBlock Text="Login" Grid.Row="1" Grid.ColumnSpan="2" FontSize="22" HorizontalAlignment="Center" VerticalAlignment="Center"/>  <TextBlock Text="Password" Grid.Row="2" Grid.ColumnSpan="2" FontSize="22" HorizontalAlignment="Center" VerticalAlignment="Center"/>  <Button Content="Вход" Grid.Column="1" Grid.Row="3" FontSize="22" Click="Button\_Click"/>  <Button Content="Выход" Grid.Column="3" Grid.Row="3" FontSize="22"/>  </Grid>  </Window> |
| Результат |
|  |
| Обработка событий |
| private void Button\_Click(object sender, RoutedEventArgs e)  {  DB\_Class.Users\_ID = \_LOG.Text;  DB\_Class.Password = \_PAS.Password;  DB\_Class.ConnStr = string.Format(DB\_Class.ConnStr, DB\_Class.Users\_ID, DB\_Class.Password);  DB\_Class db\_class = new DB\_Class();  try  {  db\_class.conn.Open();  Visibility = Visibility.Hidden;  MainWindow main = new MainWindow();  main.Show();  }  catch  {  MessageBox.Show("Не верный логин или пароль!", DB\_Class.App\_Name);  \_LOG.Focus();  }  finally  {  db\_class.conn.Close();  \_LOG.Clear();  \_PAS.Clear();  }  } |

1. ВЫВОД

Я получил навыки, на основании проведённого анализа предметной области, а именно в проведении нормализации, приведение ненормализованной формы к 3НФ, применение свойств данных, которые подлежат хранению в базе данных к нормализации отношений, приведении отношений к структурированному виду, применение принципов и механизмов межтабличных связей.

1. СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ

[Пример этапов практической](file:///\\ACER-AN\Users\logge\Desktop\база%20данных\3\Практическая%20работа%20№%202.docx) (Word, [задание](https://drive.google.com/file/d/1zYxKsKm5Ab-BnWJW_s2Mf3UT1uGUmMb6/view?usp=drive_web&authuser=1), [шаблон отчёта](https://drive.google.com/file/d/1Kc04FsjOh3HGFh_ZDjBzIs9-qh3YjYNP/view?usp=drive_web&authuser=1).