

#8 Разработка приложений WPF

Зачем делать эту лабораторную работу?

1. Чтобы научиться создавать приложения с использованием подключения к базе данных
2. Чтобы освоить инструменты технологии ADO.NET
3. Чтобы потренироваться в написании запросов на выборку и запросов модификации данных

Что нужно делать?

1. Подключиться к SQL Server и проверить обеспечение целостности данных в базе данных, созданной в предыдущих работах.

В качестве примера представлены таблицы «Группы» и «Студенты»:

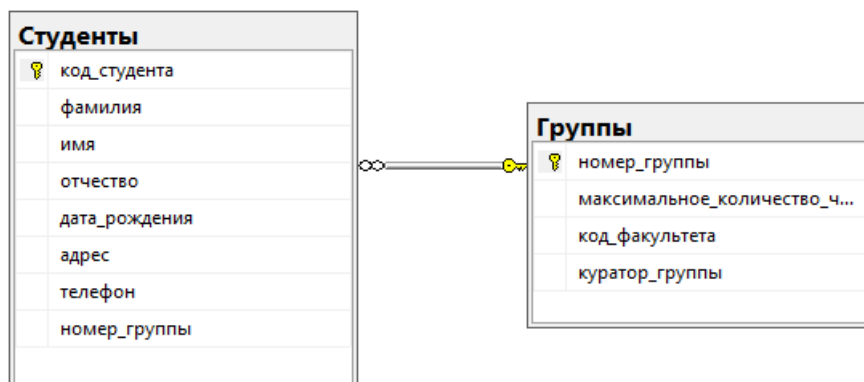
Create table Группы

```
(  
номер_группы nvarchar(10) primary key,  
максимальное_количество_человек int,  
код_факультета int,  
куратор_группы nvarchar(500))
```

Create table Студенты

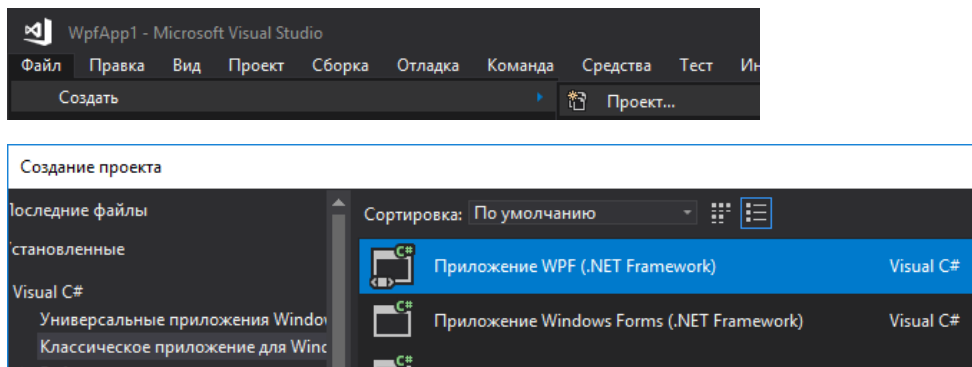
```
(код_студента int primary key,  
фамилия nvarchar(200),  
имя nvarchar(200),  
отчество nvarchar(200),  
дата_рождения date,  
адрес nvarchar(500),  
телефон nvarchar(12),  
номер_группы nvarchar(10),  
foreign key (номер_группы) references Группы (номер_группы));
```

Обеспечение целостности данных можно увидеть на схеме данных:



2. Создать приложение WPF

Выберите в меню Файл -> Создать проект в разделе Классическое приложение для Windows – Приложение WPF



3. Создать строку подключения к базе данных SQL Server

Для этого необходимо отредактировать файл App.config. Подключение к базе данных «Колледж» сервера с именем lab215-16 будет выглядеть следующим образом:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.1" />
  </startup>
  <connectionStrings>
    <add name="SQLServerConnection"
      connectionString="Data Source=lab215-16;Initial Catalog=Колледж;Integrated
Security=True"
      providerName="System.Data.SqlClient"/>
  </connectionStrings>
</configuration>
```

4. Разработать интерфейс приложения

Для приведенного ниже задания можно в конструкторе разместить следующие элементы управления, либо сделать разметку в XAML.

Для программы, которая выводит все переполненные группы, в которых количество студентов больше максимального, и предлагает варианты их перевода в другие группы, можно в конструкторе разместить следующие элементы управления, либо сделать разметку в XAML:

```
<Window x:Class="WpfApp1.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:WpfApp1"
  mc:Ignorable="d"
  Title="MainWindow" Height="450" Width="800" Loaded="Window_Loaded">
  <Grid>
    <ListBox x:Name="listStudents" HorizontalAlignment="Left" Height="172"
      Margin="88,169,0,0" VerticalAlignment="Top" Width="147"/>
    <ListBox x:Name="listChangeGroups" HorizontalAlignment="Left" Height="172"
      Margin="271,169,0,0" VerticalAlignment="Top" Width="147"/>
    <Button Content="Переполненные группы" HorizontalAlignment="Left"
      Margin="88,126,0,0" VerticalAlignment="Top" Width="147" Height="28"
      Click="Button_Click_2"/>
    <Button x:Name="optionChange" Content="Варианты перевода"
      HorizontalAlignment="Left" Margin="271,126,0,0" VerticalAlignment="Top" Width="147"
      Height="28" Click="Button_Click"/>
    <Button Content="Перевести" HorizontalAlignment="Left" Margin="440,223,0,0"
      VerticalAlignment="Top" Width="179" Height="33" Click="Button_Click_1"/>
  </Grid>
</Window>
```



```

                                where st.номер_группы = gr.номер_группы");
SqlDataReader r = cmd.ExecuteReader();
listGroups.Items.Clear();
while (r.Read())
    listGroups.Items.Add(String.Format("{0}", r.GetValue(0).ToString()));
r.Close();

}

```

Последующие действия в приложении реализуются следующим кодом:

```

private void listGroups_MouseDoubleClick(object sender,
System.Windows.Input.MouseButtonEventArgs e)
{
    studentCode = new List<int>();
    SqlCommand cmd = cnn.CreateCommand();
    cmd.CommandText = @"Select код_студента, фамилия from студенты
                                where номер_группы =
'" + listGroups.SelectedItem.ToString() + "'";
    MessageBox.Show(cmd.CommandText);
    SqlDataReader r = cmd.ExecuteReader();
    listStudents.Items.Clear();
    while (r.Read())
    {
        listStudents.Items.Add(String.Format("{0}. {1}",
r.GetValue(0).ToString(), r.GetValue(1).ToString()));
        studentCode.Add(int.Parse(r.GetValue(0).ToString()));
    }
    r.Close();
    listStudents.Visibility = Visibility.Visible;
    listGroups.Visibility = Visibility.Hidden;
}

private void Button_Click(object sender, RoutedEventArgs e)
{
    SqlCommand cmd = cnn.CreateCommand();
    cmd.CommandText = @"Select Номер_группы from Группы gr
                                where максимальное_количество_человек >
                                (Select count(*) from Студенты st
                                where st.номер_группы = gr.номер_группы)";
    MessageBox.Show(cmd.CommandText);
    SqlDataReader r = cmd.ExecuteReader();
    listChangeGroups.Items.Clear();
    while (r.Read())
        listChangeGroups.Items.Add(String.Format("{0}",
r.GetValue(0).ToString()));
    r.Close();
}

private void Button_Click_1(object sender, RoutedEventArgs e)
{
    SqlCommand cmd = cnn.CreateCommand();
    cmd.CommandText = @"Update Студенты Set номер_группы = '" +
listChangeGroups.SelectedItem.ToString() +
        @"'" where код_студента = " + studentCode[listStudents.SelectedIndex];

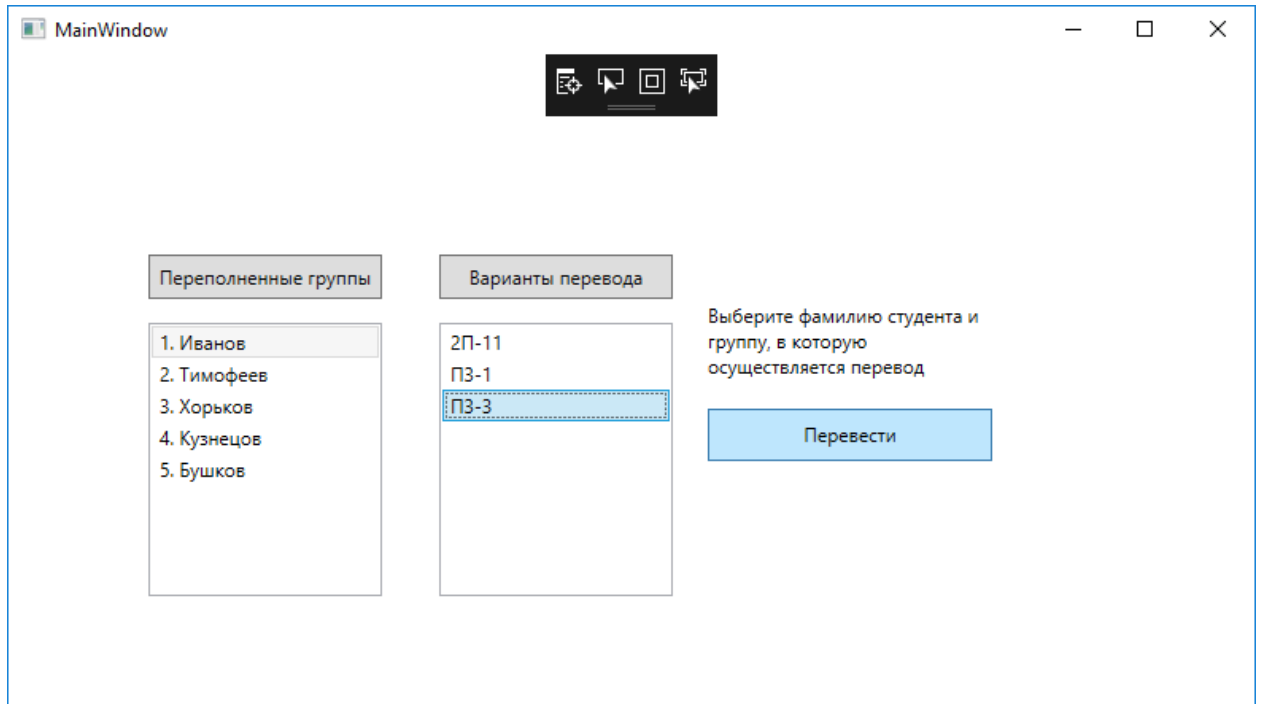
    cmd.ExecuteNonQuery();
    Button_Click(this, e);
    listGroups_MouseDoubleClick(this, null);
}

private void Button_Click_2(object sender, RoutedEventArgs e)

```

```
{
    Window_Loaded(this, null);
}
```

Результат работы приложения:



Как узнать, что всё выполнено?

Проверьте пункты в этом чек-листе:

- ☐ В базе данных обеспечена целостность данных
- ☐ Таблицы заполнены тестовыми данными
- ☐ Приложение работает без ошибок
- ☐ Приложение работает с данными из базы данных
- ☐ Работа приложения протестирована на различных наборах данных
- ☐ В приложении отсутствуют обработки данных, которые можно сделать запросами
- ☐ В приложении предусмотрена обработка исключений

Варианты заданий:

1 вариант:

Создать приложение, осуществляющее размещение товаров по свободным складам. Программа должна показывать, сколько упаковок товара находится на каждом складе, и в случае переполнения, предлагать варианты оптимизации хранения товара. Нельзя распределять товар на другие переполненные склады. Преимуществом для распределения пользуются те склады, на которых уже есть подобные товары.

2 вариант:

Создать программу, с помощью которой можно принимать товар. Информация о товаре находится в накладных, хранящихся в файле. Структура накладных произвольная. Весь товар должен автоматически приниматься на первый свободный склад. Если места на одном складе недостаточно, происходит разделение по нескольким складам, обязательно информируя об этом пользователя.

3 вариант:

Написать программу, контролирующую срок годности товаров, размещенных на складах. Программа должна запрашивать у пользователя период отчета по сроку годности и предоставлять соответствующую информацию. В случае обнаружения просроченного товара программа предлагает варианты списания товара со складов. Если срок годности товара выходит за рамки отчетного периода, списывать товар запрещается.

4 вариант:

Написать программу, которая определяет самые востребованные товары среди выбранной возрастной категории клиентов. Список этих популярных товаров должен быть сохранен в файл вместе с адресами клиентов выбранного возраста для последующей возможной рассылки в рекламных целях.

5 вариант:

Создать приложение, позволяющее создавать заказы. Необходимо предоставить пользователю возможность самостоятельно комплектовать заказ, выбирая вид товара, количество и упаковку. Также необходимо предусмотреть возможность изменения или дополнения содержимого заказа, если для него еще не прошел срок сборки. Нельзя добавлять в заказ две позиции одного и того же товара, в этом случае следует изменять количество.

6 вариант:

Создать приложение, которое выводит информацию о текущих заказах, которые находятся на этапе сборки, и распределяет их между сотрудниками. Каждый сборщик, может выбрать себе заказ с количеством упаковок не более 100. При этом большие заказы могут делиться на несколько сборщиков. Вся информация по сборке товаров записывается в текстовый файл.

7 вариант:

Составить программу распределения нагрузки среди преподавателей. При выборе преподавателя на форме должен отображаться список возможных не занятых дисциплин. Каждый преподаватель может вести не более пяти предметов. Количество пар в неделю по всем дисциплинам для преподавателя не должно превышать 32 часов.

8 вариант:

Составить программу составления расписания. Необходимо обеспечить выбор групп, дисциплин, преподавателей и лаборатории по дням недели на определенных парах. При составлении учитывать занятость лабораторий и преподавателей. Одна группа может заниматься в разных лабораториях с разными преподавателями только на некоторых предметах.

9 вариант:

Составить программу корректировки расписания по загруженности лабораторий. Программа должна выводить информацию обо всех случаях, когда в лаборатории занимается группа, в

которой число студентов больше посадочных мест и предлагать другие варианты проведения занятий.

10 вариант:

Составить программу корректировки нагрузки. При увольнении преподавателя программа должна предлагать варианты распределения предметов среди других преподавателей. Необходимо учитывать сначала нагрузку штатных преподавателей, и если свободных часов нет, предлагать преподавателям совместителям.

11 вариант:

Написать программу приема преподавателей совместителей на вакантные дисциплины. На форме обеспечить ввод необходимых данных и возможность выбора дисциплин с указанием количества часов в неделю. При выборе нескольких дисциплин, количество часов автоматически пересчитывается.

12 вариант:

Составить программу корректировки расписания. Если для какой-либо группы в расписание было поставлено больше 4 пар, программа должна предлагать варианты переноса занятия на другие дни. При этом нужно учитывать занятость лабораторий и преподавателей.

13 вариант:

Написать программу распределения интерактивного оборудования по лабораториям. В первую очередь в списке лабораторий должны выводиться те, которые находятся на кафедрах с большим числом студентов. При выборе лаборатории в таблицу базы данных должны вноситься соответствующие изменения.

14 вариант:

Составить программу корректировки расписания в случае обнаружения «окон». Если для группы в расписании есть свободные пары между другими занятиями, программа должна предлагать варианты переноса занятий с учетом занятости лабораторий и преподавателей.

15 вариант:

Составить программу корректировки расписания. В случае наложения пар для одной группы или одного преподавателя, программа должна предлагать варианты переноса занятий. При этом следует учитывать занятость лабораторий и преподавателей.

16 вариант:

Создать программу, которая проводит автоматическое зачисление студентов в группы. Зачисление происходит в случайную группу выбранного факультета. Вероятность зачисления в группы с меньшим количеством студентов должна быть выше. Зачисление в переполненные группы не допускается.

17 вариант:

Написать приложение, которое формирует заказ на самые популярные книги. В случае если более 25% книг были утеряны, автор, название книги и издательство добавляются в заказ, который сохраняется в текстовом файле. Добавлять в один заказ больше одной позиции для одной книги запрещается.

18 вариант:

Создать программу, которая для всех пользователей библиотеки, не вернувших вовремя книги, назначает штраф. Сумма штрафа должна варьироваться в зависимости от срока задержки, состояния книги. Если книга утеряна, сумма штрафа вводится работником библиотеки.

19 вариант:

Создать программу, которая составляет статистический отчет по количеству пользователей, читающих книги различных авторов и выводит периодичность посещения библиотеки для различных читателей. На форме необходимо обеспечить ввод данных об авторах и пользователях библиотеки.

20 вариант:

Создать программу, осуществляющую прием читателей в библиотеку, и исключение их в случае систематических нарушений правил пользования библиотекой. Исключение происходит только в том случае, если читатель вернул все книги или ему был назначен штраф.