ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ РЕСПУБЛИКИ ХАКАСИЯ

«ХАКАССКИЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

Специальность 09.02.07

«Информационные системы и программирование»

ПРАКТИЧЕСКОЕ Занятие №21-22

тема: Реализация изменения и удаления данных

Руководитель:

\_\_\_\_\_\_\_\_\_\_\_\_ Д.С.Черкашин

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, дата)

Выполнил:

Студент группы ИС(ПРО)-42

\_\_\_\_\_\_\_\_\_\_\_\_ М.А.Ламаев

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Абакан 2023 г.

**Описание задания**

1. Реализация редактирования и удаления данных

**Листинг:**

Листинг 1 — Создание таблицы, события кнопки

<DataGrid x:Name="MedicalRecordsDataGrid"  
 AutoGenerateColumns="False"  
 IsReadOnly="True"  
 GridLinesVisibility="All"  
 BorderThickness="1"  
 BorderBrush="Gray"  
 x:DataType="models:medical\_records"  
 Margin="0,-150,0,0"  
 Background="White"  
 MaxHeight="200"  
   
 SelectionChanged="MedicalRecordsDataGrid\_OnSelectionChanged" >   
 <DataGrid.Columns>  
 <DataGridTextColumn Header="Фамилия пациента" Binding="{Binding patients}"/>  
 <DataGridTextColumn Header="Фамилия доктор" Binding="{Binding doctors}"/>  
 <DataGridTextColumn Header="Диагноз" Binding="{Binding diagnosis}"/>  
 <DataGridTextColumn Header="Состояние" Binding="{Binding state}"/>  
 </DataGrid.Columns>  
</DataGrid>  
<StackPanel HorizontalAlignment="Left" VerticalAlignment="Center" Margin="0,150,0,0" Orientation="Horizontal" >  
 <Button x:Name="Apdate" Content="Обновить" Foreground="White" MinHeight="45" VerticalContentAlignment="Center" Click="Apdate\_OnClick" />  
 <Button x:Name="AddMedicalRecord" Content="Добавить запись" Foreground="White" MinHeight="45" VerticalContentAlignment="Center" Margin="5,0,0,0" Click="AddMedicalRecord\_OnClick" />  
 <Button x:Name="EditMedicalRecord" Content="Изменить запись" Foreground="White" MinHeight="45" VerticalContentAlignment="Center" Margin="5,0,0,0" Click="EditMedicalRecord\_OnClick" />  
 <Button x:Name="RemoveMedicalRecord" Content="Удалить запись" Foreground="White" MinHeight="45" VerticalContentAlignment="Center" Margin="5,0,0,0" Click="RemoveMedicalRecord\_OnClick" />  
 <Button x:Name="Close" Content="Закрыть" Foreground="White" MinHeight="45" VerticalContentAlignment="Center" Margin="5,0,0,0" Click="Close\_OnClick" />  
</StackPanel>

В этом коде создается таблица в котором будет происходить удаление и изменение. Добавление кнопок для удаления и изменения. Создается событие SelectionChange

Листинг 2 — метод MedicalRecordsDataGrid\_OnSelectionChanged

private void MedicalRecordsDataGrid\_OnSelectionChanged(object? sender, SelectionChangedEventArgs e)  
{  
 DeleteMedicalRecords = MedicalRecordsDataGrid.SelectedItem as medical\_records;  
 DataContext = MedicalRecordsDataGrid.SelectedItem as medical\_records;  
}

Здесь были созданы переменые, в которые передается запись выбранная в таблице.

Листинг 3 - метод RemoveMedicalRecord\_OnClick

private async void RemoveMedicalRecord\_OnClick(object? sender, RoutedEventArgs e)  
{  
 var box = MessageBoxManager.GetMessageBoxStandard("Вы уверены", "Вы уверены что хотите удалить запись?",  
 ButtonEnum.YesNo);  
 var result = await box.ShowAsync();  
 if (result == ButtonResult.No) return;  
 if (DeleteMedicalRecords == null) return;  
 if (result == ButtonResult.Yes)  
 {  
 using (var connection = new MySqlConnection(\_connectionString.ConnectionString))  
 {  
 connection.Open();  
 using (var command = connection.CreateCommand())  
 {  
 command.CommandText = "DELETE FROM medical\_records WHERE id = @id";  
 command.Parameters.AddWithValue("@id", DeleteMedicalRecords.id);  
 var rowsCount = command.ExecuteNonQuery();  
 }  
  
 UpdateMedicalRecords();  
 connection.Close();  
 }  
 }  
}

В этом метод реализовано удаление выбранных данных. И условие удалять если пользователь нажмет «Yes», иначе метод ничего не будет делать. Так же ничего не произойдет если пользователь ничего не выберет

Листинг 4 — метод EditMedicalRecord\_OnClick

private void EditMedicalRecord\_OnClick(object? sender, RoutedEventArgs e)  
{  
 if (sender is Button button)  
 {  
 if (button.DataContext is medical\_records MedicalRecords)  
 {  
 new AddMedicalRecords(MedicalRecords).ShowDialog(this);  
 UpdateMedicalRecords();  
 }  
 }  
}

Переменная которая содержит выбранную запись передается в окно Добавления/изменения.

Листинг 5 - инцилизация

public partial class AddMedicalRecords : Window  
{  
 private medical\_records NewMedicalRecords { get; set; }  
 private List<medical\_records> MedicalRecordsList { get; set; }  
 private List<patients> NewPatientsList { get; set; }  
 private List<doctors> NewDoctorsList { get; set; }  
 private List<diagnosis> NewDiagnosisList { get; set; }  
 private List<state> NewStateList { get; set; }

Продолжение листинга 5

public MySqlConnectionStringBuilder \_connectionString;

private bool IsUpdated { get; set; }

public AddMedicalRecords( medical\_records MedicalRecords = null)

{  
 InitializeComponent();  
 MedicalRecordsList = new List<medical\_records>();  
 NewPatientsList = new List<patients>();  
 NewDoctorsList = new List<doctors>();  
 NewDiagnosisList = new List<diagnosis>();  
 NewStateList = new List<state>();  
 \_connectionString = new MySqlConnectionStringBuilder()  
 {  
 Server = "10.10.1.24",  
 Database = "pro3",  
 UserID = "user\_01",  
 Password = "user01pro"  
 // Server = "localhost",  
 // Database = "pr3",  
 // UserID = "root",  
 // Password = "123456"  
 };  
 PatientsComboBox = this.Find<ComboBox>("PatientsComboBox");  
 DoctorsComboBox = this.Find<ComboBox>("DoctorsComboBox");  
 DiagnosisComboBox = this.Find<ComboBox>("DiagnosisComboBox");  
 StateComboBox = this.Find<ComboBox>("StateComboBox");  
 UpdatePatientsComboBox();  
 UpdateDoctorsComboBox();  
 UpdateDiagnosisComboBox();  
 UpdateStateComboBox();  
   
 if (MedicalRecords == null)  
 {  
 NewMedicalRecords = new medical\_records();  
 IsUpdated = false;  
 }  
 else  
 {  
 NewMedicalRecords = MedicalRecords;  
 PatientsComboBox.SelectedItem = NewPatientsList.FirstOrDefault(g => g.lastName == NewMedicalRecords.patients);  
 DoctorsComboBox.SelectedItem = NewDoctorsList.FirstOrDefault(g => g.lastName\_ == NewMedicalRecords.doctors);  
 DiagnosisComboBox.SelectedItem = NewDiagnosisList.FirstOrDefault(g => g.name\_diagnosis == NewMedicalRecords.diagnosis);  
 StateComboBox.SelectedItem = NewStateList.FirstOrDefault(g => g.name\_state == NewMedicalRecords.state);  
 IsUpdated = true;  
 }  
   
 }

IsUpdated свойство типа bool, которое будет сообщать остальному коду о том, открыта текущая страница для добавления или для редактирования. Если переданная переменная с выбранной записью есть то окно будет работать в режиме изменения, иначе в режиме добавления

Листинг 6 — метод AddData()

private void AddData()  
{  
 using (var c = new MySqlConnection(\_connectionString.ConnectionString))  
 {  
 c.Open();  
 using (var cmd = c.CreateCommand())  
 {

Продолжение листинга 6

cmd.CommandText =

"INSERT INTO medical\_records (patients,doctors,diagnosis,state) " +  
 "VALUES (@patients,@doctors,@diagnosis,@state)";  
  
 cmd.Parameters.AddWithValue("@patients", ((PatientsComboBox.SelectedItem) as patients).id);

cmd.Parameters.AddWithValue("@doctors", ((DoctorsComboBox.SelectedItem) as doctors).id);  
 cmd.Parameters.AddWithValue("@diagnosis", ((DiagnosisComboBox.SelectedItem) as diagnosis).id);  
 cmd.Parameters.AddWithValue("@state", ((StateComboBox.SelectedItem) as state).id);  
  
 var rowsCount = cmd.ExecuteNonQuery();  
 if (rowsCount == 0)  
 {  
 var box = MessageBoxManager.GetMessageBoxStandard("Ошибка", "Не удалось выполнить добавление!",  
 ButtonEnum.Ok);  
 box.ShowAsync();  
 }  
 else  
 {  
 var box1 = MessageBoxManager.GetMessageBoxStandard("Успех", "Удалось выполнить добавление!",  
 ButtonEnum.Ok);  
 box1.ShowAsync();  
 }  
  
 c.Close();  
 }  
  
 this.Close();  
 }  
}

В этом методе при нажатии на кнопку совершается добавление в базу.

Листинг 7 — метод UpdateData()

private void UpdateData()  
{  
 using (var c = new MySqlConnection(\_connectionString.ConnectionString))  
 {  
 c.Open();  
 using (var cmd = c.CreateCommand())  
 {  
 cmd.CommandText = "UPDATE medical\_records " +   
 "SET patients = @patients, doctors = @doctors, diagnosis = @diagnosis, state = @state " +   
 "WHERE id = @id;";  
   
 cmd.Parameters.AddWithValue("@id", NewMedicalRecords.id);  
 cmd.Parameters.AddWithValue("@patients", ((PatientsComboBox.SelectedItem) as patients).id);  
 cmd.Parameters.AddWithValue("@doctors", ((DoctorsComboBox.SelectedItem) as doctors).id);  
 cmd.Parameters.AddWithValue("@diagnosis", ((DiagnosisComboBox.SelectedItem) as diagnosis).id);  
 cmd.Parameters.AddWithValue("@state", ((StateComboBox.SelectedItem) as

Продолжение листинга 7

state).id);  
  
 var rowsCount = cmd.ExecuteNonQuery();  
 if (rowsCount == 0)  
 {  
 var box = MessageBoxManager.GetMessageBoxStandard("Ошибка", "Не удалось выполнить изменение!",  
 ButtonEnum.Ok);

box.ShowAsync();

}  
 else  
 {  
 var box1 = MessageBoxManager.GetMessageBoxStandard("Успех", "Удалось

выполнить изменение!",  
 ButtonEnum.Ok);  
 box1.ShowAsync();  
 }  
  
 c.Close();  
 }  
  
 this.Close();  
 }  
}

Продолжение листинга 8 — обработчик кнопки «Сохранить»

private void Add\_medical\_records\_OnClick(object? sender, RoutedEventArgs e)  
{  
 if (IsUpdated)  
 {  
 UpdateData();  
 }  
 else  
 {  
 AddData();  
 }  
}

Если переменная IsUpdated в режиме true, то кнопка «Сохранить» изменяет запись, иначе добавляет.

**Скриншоты**

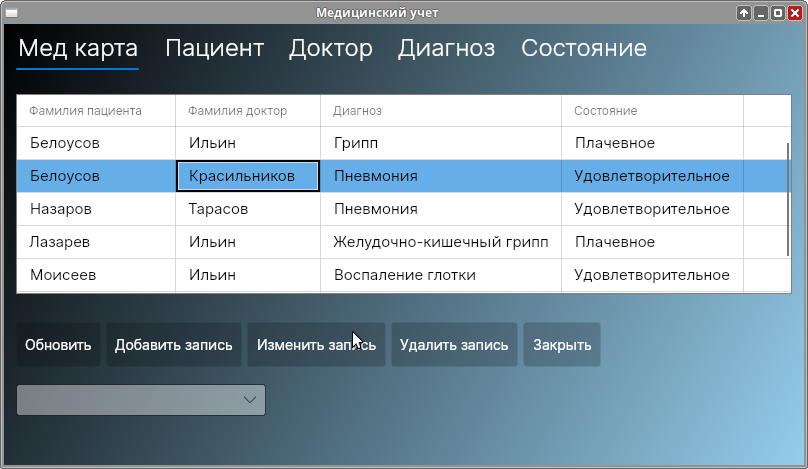


Рисунок 1 – Изменение записей

Для того чтобы изменить запись, необходимо щелкнуть мышью по записе в таблице, которая вас интересует.

Затем нажать на кнопку «Изменить запись».

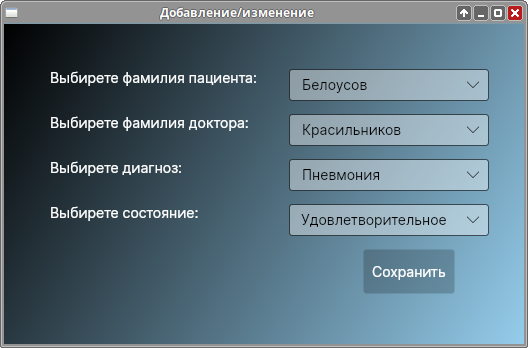


Рисунок 2 – Окно изменения

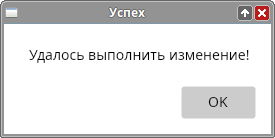


Рисунок 3 - Сообщение

Откроется окно изменение.

Выбираем в списках данные о пациенте, которые хотим заменить.

Сохраняем.

Если все получилось выведется сообщения «Успех».

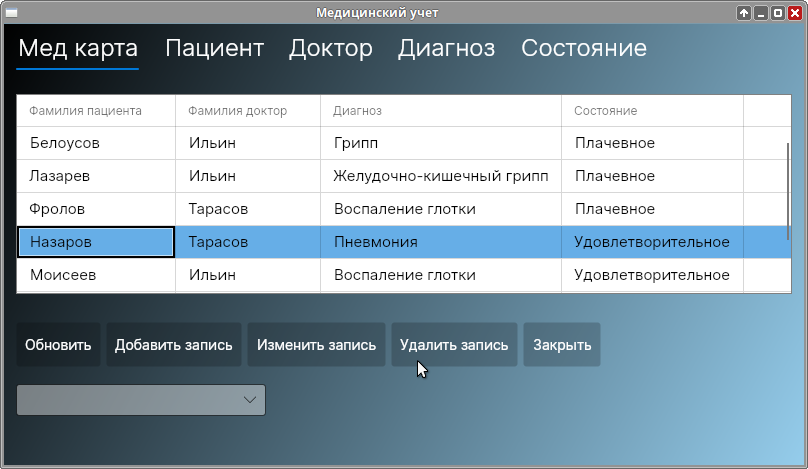


Рисунок 4 – Удаление записей

Для того чтобы удалить запись, необходимо щелкнуть мышью по записе в таблице, которая вас интересует.

Затем нажать на кнопку «Удалить запись».

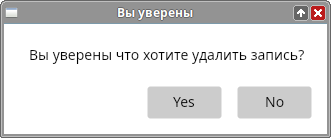


Рисунок 5 — Сообщение

Выведется сообщение, в котором нужно выбрать Yes/No