Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №10**

По дисциплине «СПП»

за 6-й семестр

Выполнил:

студент 3 курса

группы ПО-3 (1)

Афанасьев В.В.

Проверил:

Крощенко А.А.

Брест, 2021

**Цель работы:** приобрести практические навыки разработки многооконных приложений на WPF для работыс базами данных.

**Вариант:** 24

**Задание:**

На основе БД, разработанной в лабораторной работе №9, реализовать многооконное приложениеклиент, позволяющее выполнять основные операции над таблицей в БД (добавление, удаление, модификацию данных).Основные требования к приложению: *•* Для отображения выбирать таблицу с внешними ключами; *•* Осуществлять вывод основных данных в табличном представлении; *•* При выводе краткого представления записи в таблице (т.е. если выводятся не все поля), пощелчку мышкой на запись осуществлять вывод всех полей в подготовленные компоненты наформе; *•* Для всех полей, представленных внешними ключами, выводить их текстовое представлениеиз связанных таблиц (например, таблица-справочник «Времена года» содержит два поля –идентификатор и название сезона, в связанной таблице «Месяц года» есть внешний ключ натаблицу «Времена года»; в этом случае при выводе таблицы «Месяц года» нужно выводитьназвание сезона, а не его идентификатор); *•* При выводе предусмотреть упорядочивание по столбцу; *•* Реализовать простейший фильтр данных по одному-двум полям; *•* При добавлении новых данных в таблицу использовать дополнительное окно для ввода; *•* При модификации данных можно использовать ту же форму, что и для добавления, но с внесенными актуальными значениями полей; *•* При добавлении/модификации выводить варианты значений полей с внешним ключом с помощью выпадающего списка; *•* При удалении данных осуществлять удаление записи, на которой в данных момент находитсяфокус.

**Код программы:**

**MainWindow.xaml**

using System.Linq;

using System.Windows;

using spp3.Model1;

namespace spp3.View

{

/// <summary>

/// Логика взаимодействия для UpdateRacer.xaml

/// </summary>

public partial class UpdateRent : Window

{

private static Rent \_rent;

public UpdateRent(Rent rent)

{

InitializeComponent();

using RentContext context = new RentContext();

\_rent = rent;

Client.ItemsSource = context.Clients

.Select(r => r.Name)

.ToList();

Car.ItemsSource = context.Cars

.Select(c => c.Name)

.ToList();

Rate.ItemsSource = context.Rates

.Select(t => t.Name)

.ToList();

Client.SelectedItem = rent.Client.Name;

Car.SelectedItem = rent.Car.Name;

Rate.SelectedItem = rent.Rate.Name;

Name.Text = rent.Name;

}

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

{

this.DialogResult = true;

}

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

{

using RentContext context = new RentContext();

context.Rents.Remove(context.Rents.First(r => r == \_rent));

context.SaveChanges();

this.Close();

}

}

}

**UpdateRent.xaml:**

using System.Linq;

using System.Windows;

using spp3.Model1;

namespace spp3.View

{

/// <summary>

/// Логика взаимодействия для UpdateRacer.xaml

/// </summary>

public partial class UpdateRent : Window

{

private static Rent \_rent;

public UpdateRent(Rent rent)

{

InitializeComponent();

using RentContext context = new RentContext();

\_rent = rent;

Client.ItemsSource = context.Clients

.Select(r => r.Name)

.ToList();

Car.ItemsSource = context.Cars

.Select(c => c.Name)

.ToList();

Rate.ItemsSource = context.Rates

.Select(t => t.Name)

.ToList();

Client.SelectedItem = rent.Client.Name;

Car.SelectedItem = rent.Car.Name;

Rate.SelectedItem = rent.Rate.Name;

Name.Text = rent.Name;

}

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

{

this.DialogResult = true;

}

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

{

using RentContext context = new RentContext();

context.Rents.Remove(context.Rents.First(r => r == \_rent));

context.SaveChanges();

this.Close();

}

}

}

**AddCar.cs:**

using System.Linq;

using System.Windows;

using spp3.Model1;

namespace spp3.View

{

/// <summary>

/// Логика взаимодействия для AddRacer.xaml

/// </summary>

public partial class AddRent : Window

{

public AddRent()

{

InitializeComponent();

using RentContext context = new RentContext();

Client.ItemsSource = context.Clients

.Select(r => r.Name)

.ToList();

Car.ItemsSource = context.Cars

.Select(c => c.Name)

.ToList();

Rate.ItemsSource = context.Rates

.Select(t => t.Name)

.ToList();

}

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

{

this.DialogResult = true;

}

}

}

**AddRate.xaml:**

using System.Windows;

namespace spp3.View

{

/// <summary>

/// Логика взаимодействия для AddCar.xaml

/// </summary>

public partial class AddRate : Window

{

public AddRate()

{

InitializeComponent();

}

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

{

this.DialogResult = true;

}

}

}

**AddProvider.xaml:**

using System.Windows;

namespace spp3.View

{

/// <summary>

/// Логика взаимодействия для AddRace.xaml

/// </summary>

public partial class AddProvider : Window

{

public AddProvider()

{

InitializeComponent();

}

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

{

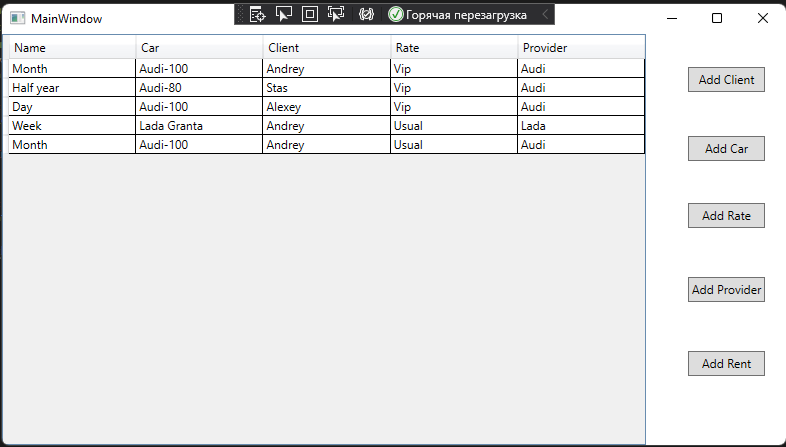
this.DialogResult = true;

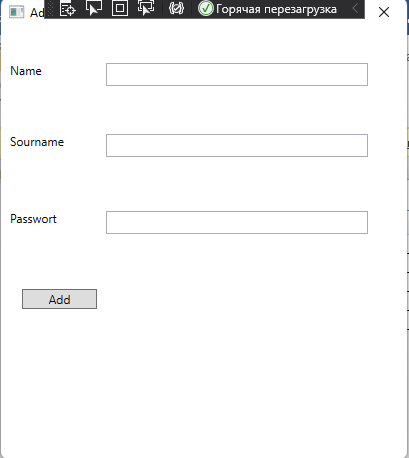
}

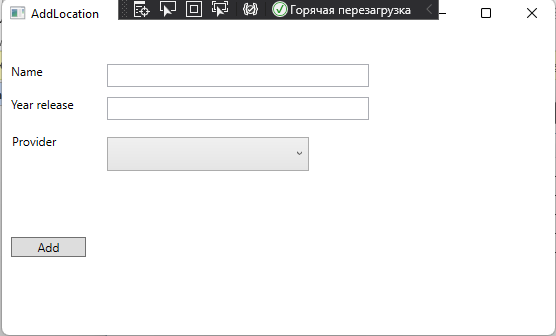
}

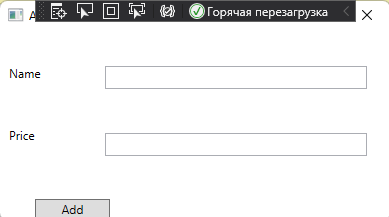
}

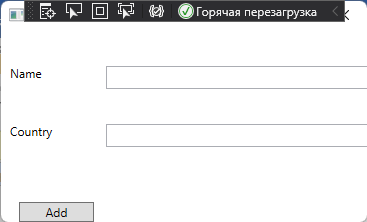
**Результаты работы:**

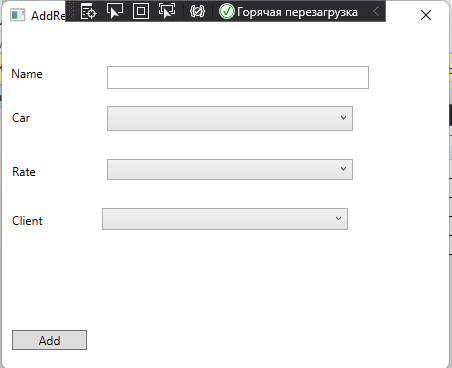
****

****

****

****

****

****

**Выводы:** в ходе выполнения лабораторной работы были приобретены практические навыки разработки многооконных приложений на WPF для работы с базами данных.