|  |  |
| --- | --- |
| MainWindow GRID  <Grid>  <Grid.ColumnDefinitions>  <ColumnDefinition Width="198\*"/>  <ColumnDefinition Width="199\*"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="57\*"/>  <RowDefinition Height="13\*"/>  </Grid.RowDefinitions>  <DataGrid Name="DGridHotels" AutoGenerateColumns="False" IsReadOnly="True" Grid.ColumnSpan="2" Grid.RowSpan="1" Grid.Row="0" Grid.Column="0">  <DataGrid.Columns>  <DataGridTextColumn Header="Статус" Binding="{Binding OrderStatus}" Width="200"></DataGridTextColumn>  <DataGridTextColumn Header="Дата доставки" Binding="{Binding OrderDeliveryDate}" Width="200"></DataGridTextColumn>  <DataGridTextColumn Header="Дата заказа" Binding="{Binding OrderDate}" Width="200"></DataGridTextColumn>  <DataGridTextColumn Header="ФИО клиента" Binding="{Binding ClientFullName}" Width="200"></DataGridTextColumn>  <DataGridTextColumn Header="ПикапПоинт" Binding="{Binding PickupPoint.Addres}" Width="200"></DataGridTextColumn>  <DataGridTemplateColumn Width="\*">  <DataGridTemplateColumn.CellTemplate>  <DataTemplate>  <Button Content="Редактировать" Click="Edit\_Click" Width="150" Height="20" Name="btnEdit"></Button>  </DataTemplate>  </DataGridTemplateColumn.CellTemplate>  </DataGridTemplateColumn>  </DataGrid.Columns>  </DataGrid>  <Button Content="Добавить запись" Name="btnAdd" Click="btnAdd\_Click" Grid.Column="0" Grid.Row="1" Width="200" Height="50" Margin="100,0,0,0"></Button>  <Button Content="Удалить запись" Name="btnDelete" Click="btnDelete\_Click" Grid.Column="1" Grid.Row="1" Width="200" Height="50" Margin="0,0,100,0"></Button>  </Grid> | CS mainwindow cs  public partial class MainWindow : Window  {  public MainWindow()  {  InitializeComponent();  DGridHotels.ItemsSource = Connect.GetContext().Order.ToList();  }  private void btnAdd\_Click(object sender, RoutedEventArgs e)  {  AddEdit gr = new AddEdit(null);  gr.Show();  this.Visibility = Visibility.Collapsed;  }  private void btnDelete\_Click(object sender, RoutedEventArgs e)  {  var hotelForRemoving = DGridHotels.SelectedItems.Cast<Order>().ToList();  if (MessageBox.Show($"Вы точно хотите удалить следующие {hotelForRemoving.Count()} элементов?", "Внимание", MessageBoxButton.YesNo, MessageBoxImage.Question) == MessageBoxResult.Yes)  {  try  {  Connect.GetContext().Order.RemoveRange(hotelForRemoving);  Connect.GetContext().SaveChanges();  MessageBox.Show("Данные удалены");  DGridHotels.ItemsSource = Connect.GetContext().Order.ToList();  }  catch (Exception ex)  {  MessageBox.Show(ex.Message.ToString());  }  }  }  private void Edit\_Click(object sender, RoutedEventArgs e)  {  AddEdit gr = new AddEdit((sender as Button).DataContext as Order);  gr.Show();  this.Visibility = Visibility.Collapsed;  }  }  } |
| AddEdit  GRID  <Grid>  <Grid.ColumnDefinitions>  <ColumnDefinition Width="163\*"/>  <ColumnDefinition Width="17\*"/>  <ColumnDefinition Width="217\*"/>  </Grid.ColumnDefinitions>  <Grid.RowDefinitions>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  <RowDefinition Height="70\*"/>  </Grid.RowDefinitions>  <TextBlock Text="Статус" FontSize="20" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="0" ></TextBlock>  <TextBlock Text="Дата доставки" FontSize="20" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="1"></TextBlock>  <TextBlock Text="Дата заказа" FontSize="20" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="2"></TextBlock>  <TextBlock Text="пикап поинта" FontSize="20" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="4"></TextBlock>  <TextBlock Text="ФИО клиента" FontSize="20" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="3"></TextBlock>  <TextBox TextAlignment="Center" FontSize="20" Text="{Binding OrderStatus}" Grid.Row="0" Grid.Column="2" VerticalAlignment="Center" Width="300" Height="40"></TextBox>  <TextBox TextAlignment="Center" FontSize="20" Text="{Binding OrderDeliveryDate}" Grid.Row="1" Grid.Column="2" VerticalAlignment="Center" Width="300" Height="40"></TextBox>  <TextBox TextAlignment="Center" FontSize="20" Text="{Binding OrderDate}" Grid.Row="2" Grid.Column="2" VerticalAlignment="Center" Width="300" Height="40"></TextBox>  <TextBox TextAlignment="Center" FontSize="20" Text="{Binding ClientFullName}" Grid.Row="3" Grid.Column="2" VerticalAlignment="Center" Width="300" Height="40"></TextBox>  <ComboBox FontSize="24" Name="ComboCountries" SelectedItem="{Binding PickupPoint}" DisplayMemberPath="Addres" Grid.Row="4" Grid.Column="3" VerticalAlignment="Center" Width="300" Height="40" ></ComboBox>  <Button Content="Сохранить" Click="Save\_Click" Width="220" Height="45" FontSize="20" Grid.Column="0" Grid.Row="8" VerticalAlignment="Center" HorizontalAlignment="Right"></Button>  <Button Content="Назад" Click="Back\_Click" Width="220" Height="45" FontSize="20" Grid.Column="2" Grid.Row="8" VerticalAlignment="Center" HorizontalAlignment="Left"></Button>  </Grid> | CS addedit  public partial class AddEdit : Window  {  private Order \_currentOrder = new Order();  public AddEdit(Order selectedOrder)  {  InitializeComponent();  if (selectedOrder != null)  {  \_currentOrder = selectedOrder;  }  DataContext = \_currentOrder;  ComboCountries.ItemsSource = Connect.GetContext().PickupPoint.ToList();  }  private void Back\_Click(object sender, RoutedEventArgs e)  {  MainWindow bc = new MainWindow();  bc.Show();  this.Visibility = Visibility.Collapsed;  }  private void Save\_Click(object sender, RoutedEventArgs e)  {  StringBuilder error = new StringBuilder();  if (string.IsNullOrWhiteSpace(\_currentOrder.OrderStatus))  error.AppendLine("Укажите название отеля");  //if (\_currentOrder.PickupPoint == null)  // error.AppendLine("Выберите пикап поинт");  if (error.Length > 0)  {  MessageBox.Show(error.ToString());  return;  }  if (\_currentOrder.OrderID == 0)  Connect.GetContext().Order.Add(\_currentOrder);  try  {  Connect.GetContext().SaveChanges();  MessageBox.Show("Информация сохранена");  }  finally  {  }  // catch (Exception ex)  //{  // MessageBox.Show(ex.Message.ToString());  // }  }  }  } |
| Connect  class Connect  {  public static RulDemkEntities bd;  public static RulDemkEntities GetContext()  {  if (bd == null)  {  bd = new RulDemkEntities();  }  return bd;  }  }  } |  |