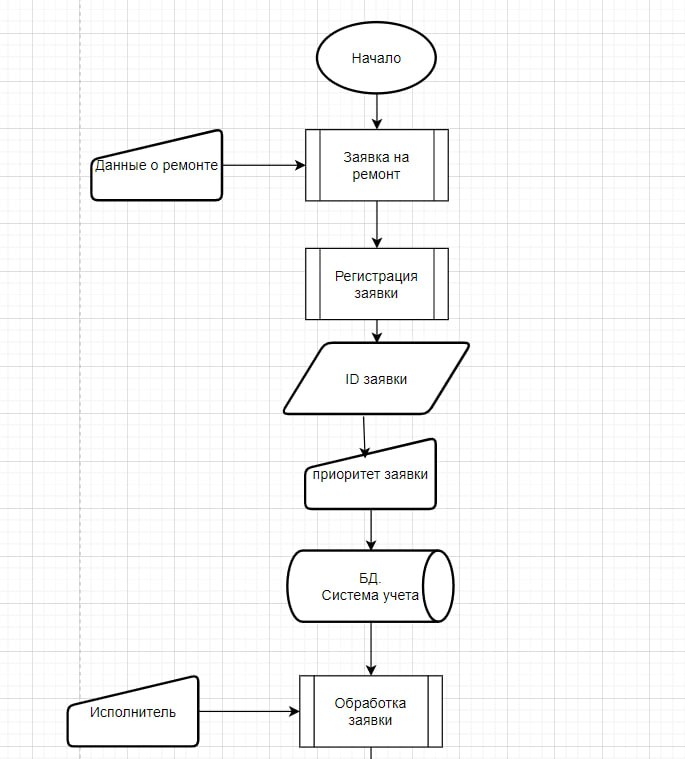
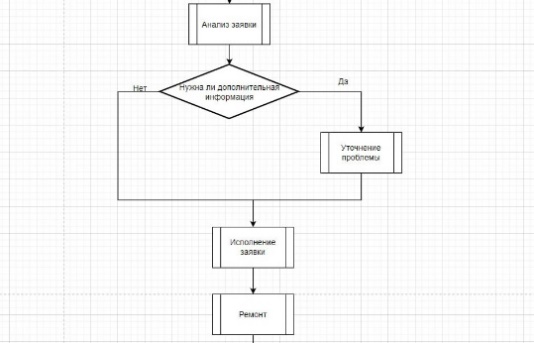
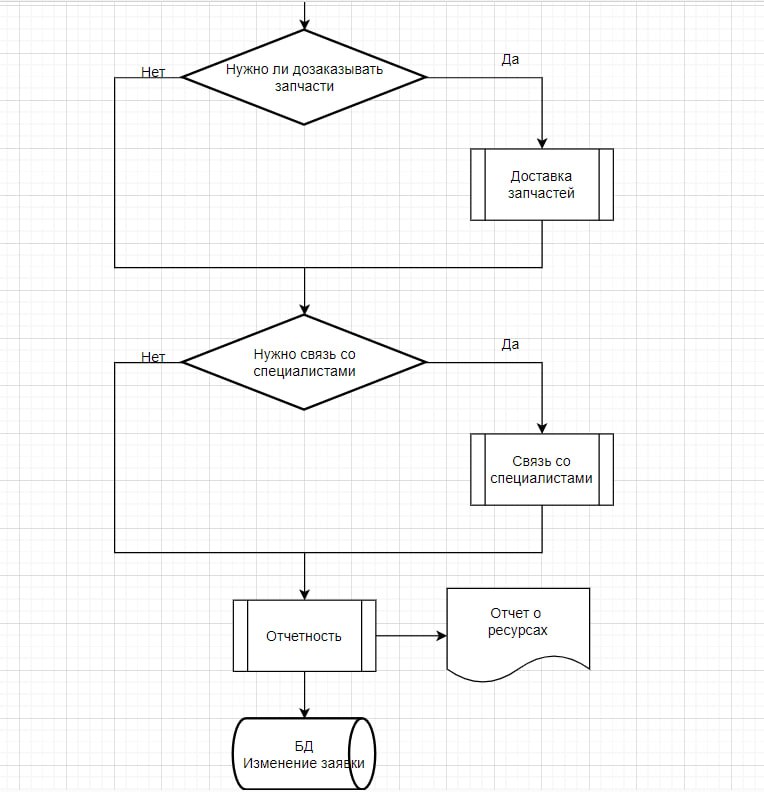
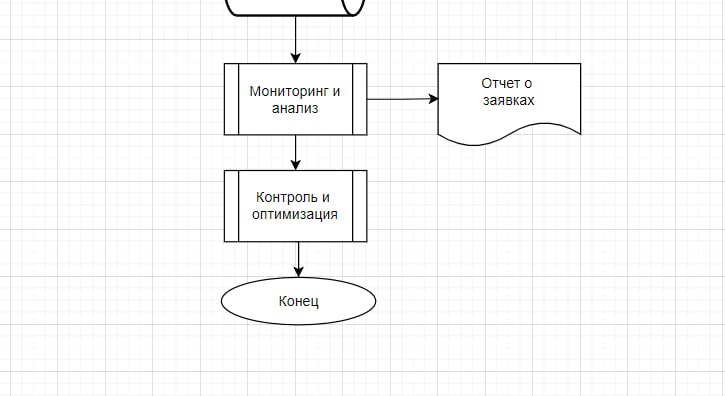
1. Блок-схема общая



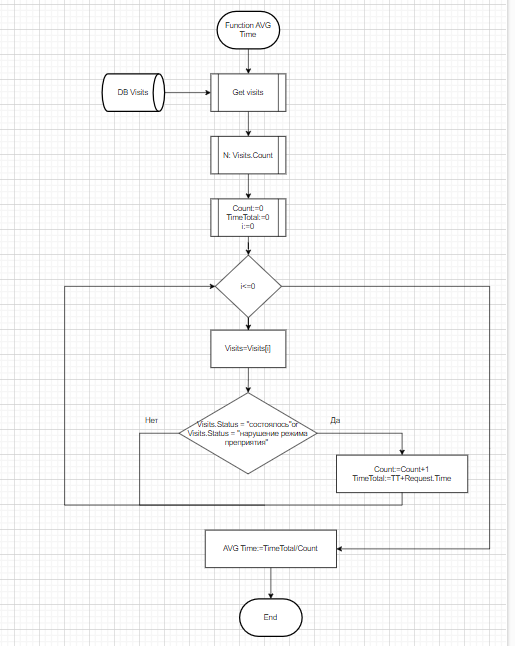






|  |  |
| --- | --- |
| Начало |  |
| Основной процесс(в описании ПР название пункта) |  |
| Данные, которые ввели с клавиатуры |  |
| Данные, которые неизвестно откуда берутся |  |
| Данные из бд, например таблица |  |
| Условие, от которого идет да нет |  |
| Отчет |  |

1. Детальный алгоритм



1. Оформление работы с **отладчиком** в окне «Авторизация» путем создания скриншотов действий с отладчиком:

* установка точки останова (клавиша F9);
* пошаговый режим работы отладчика (клавиша F10);
* просмотр всплывающей подсказки для текущих значений локальных величин при наведении курсора мыши;

окно значений локальных переменных.

Установка точки прерывания

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, веб-страница

Автоматически созданное описание

Просмотр всплывающей подсказки для текущих значений локальных величин при наведении курсора мыши

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, компьютер

Автоматически созданное описание

Пошаговый режим работы отладчика

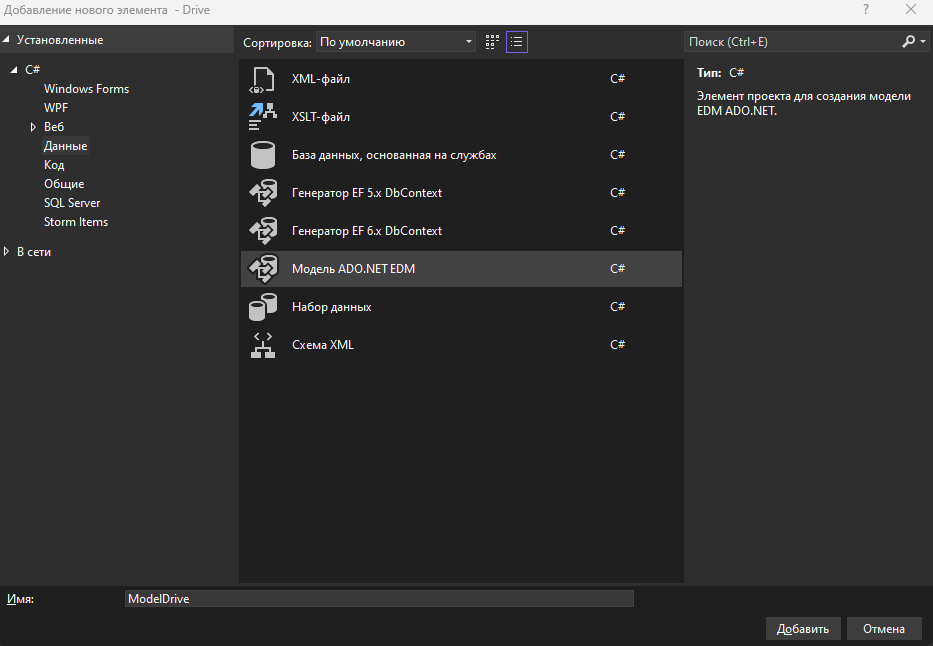
Изображение выглядит как текст, программное обеспечение, компьютер, Значок на компьютере

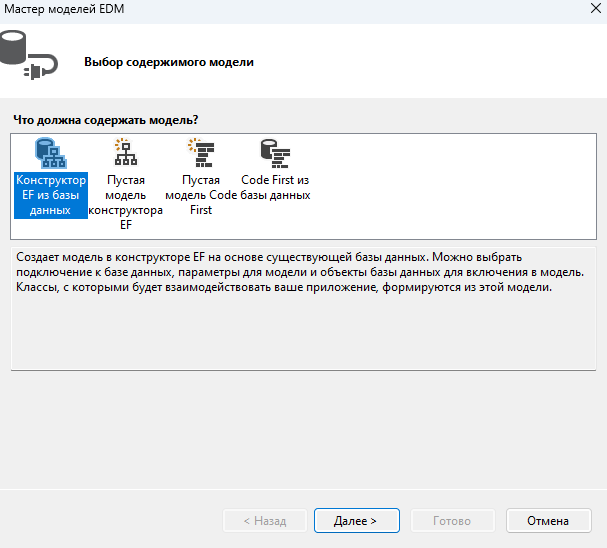
Автоматически созданное описание

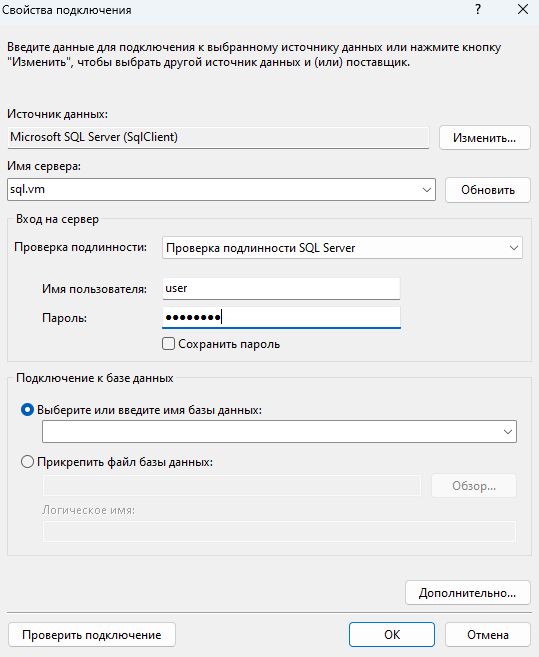
Изображение выглядит как текст, программное обеспечение, Значок на компьютере, веб-страница

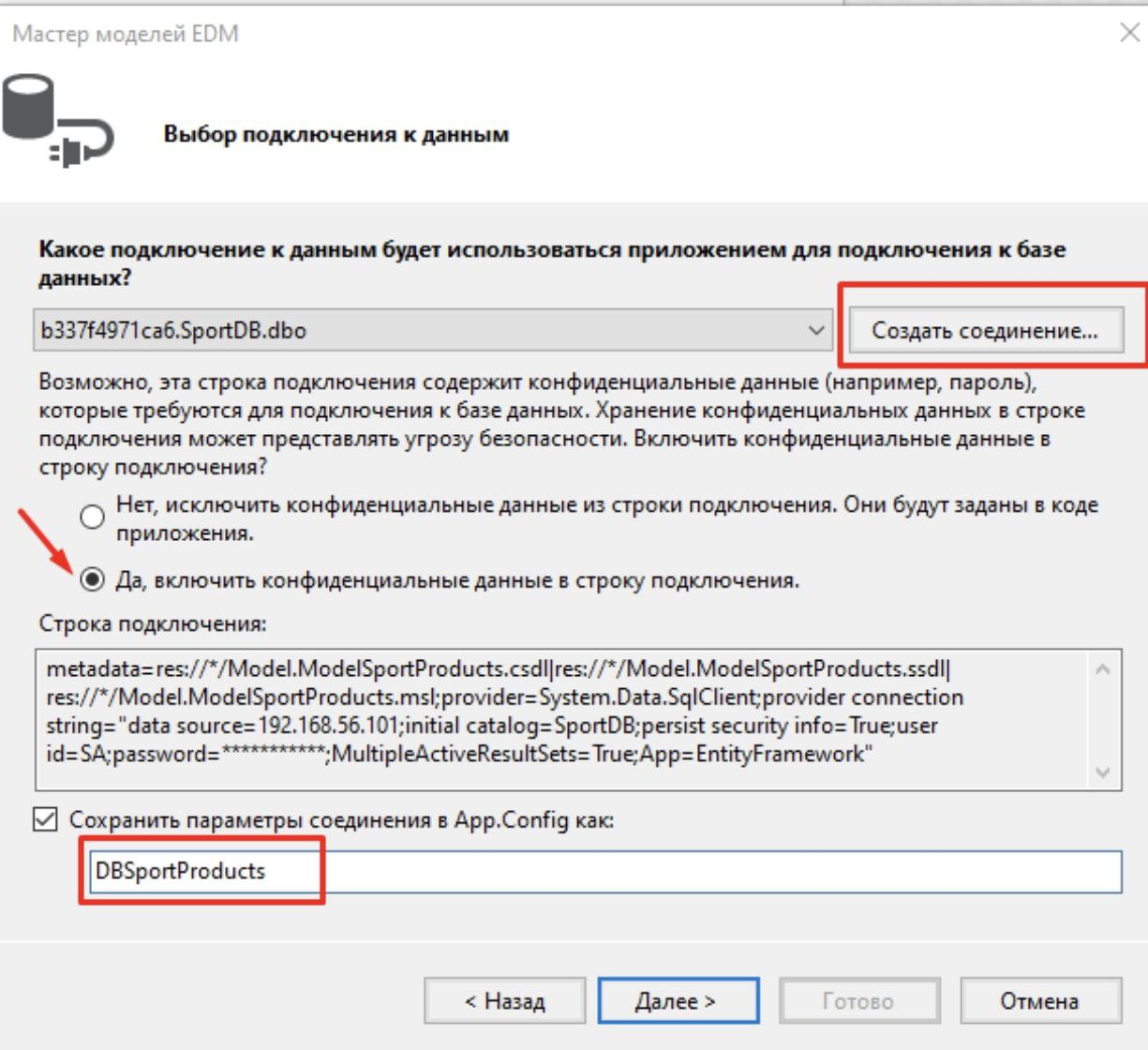
Автоматически созданное описание

1. Создание модели БД

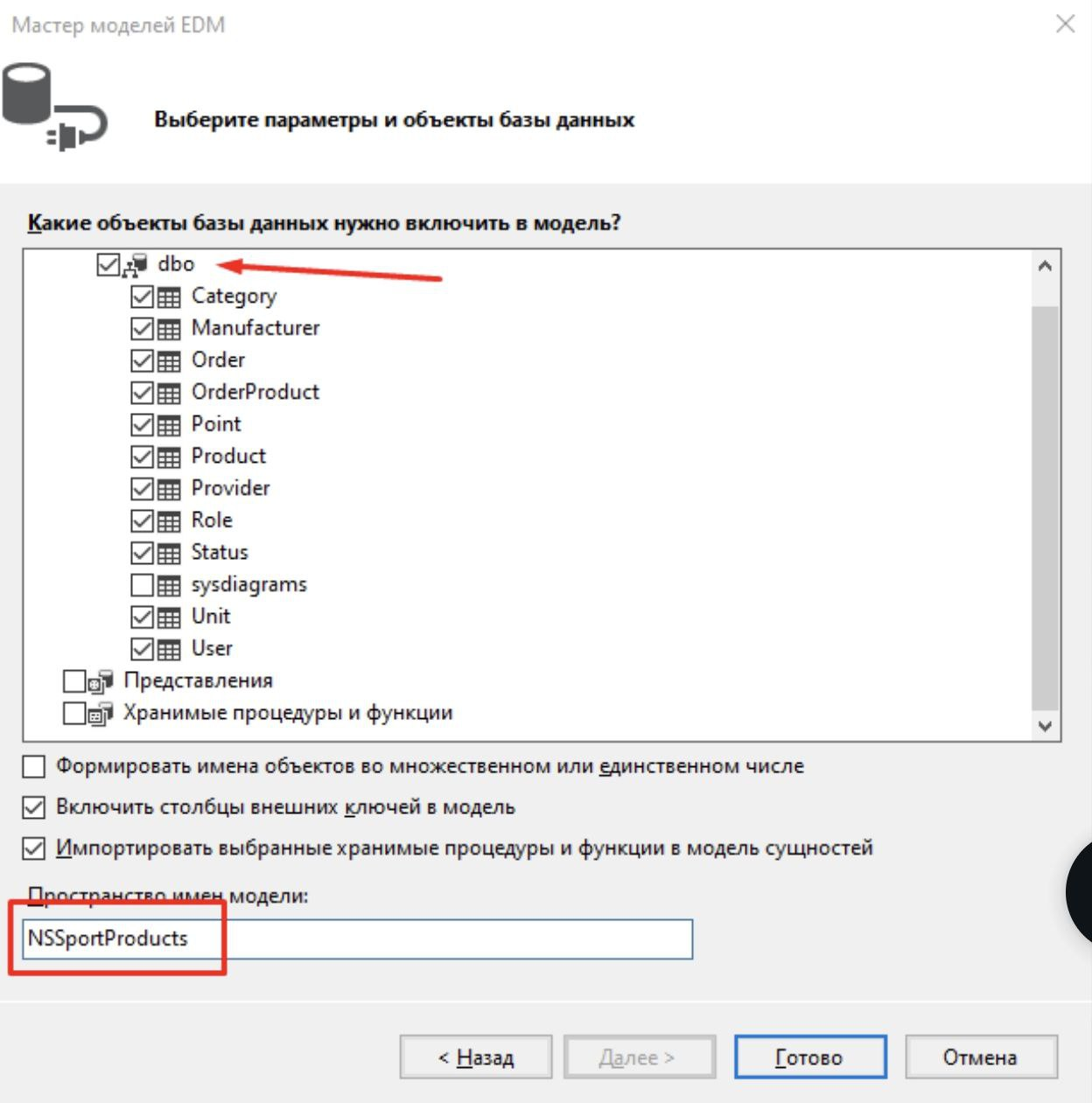






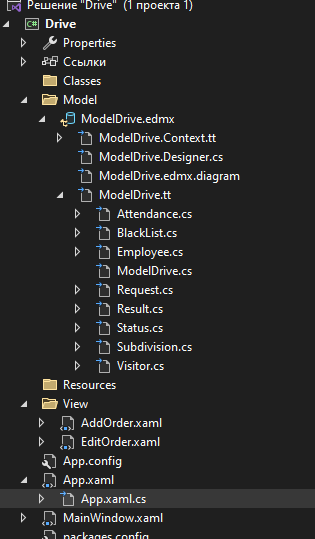








1. КОД:



* App.xaml

<Application x:Class="Drive.App"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:local="clr-namespace:Drive"

StartupUri="MainWindow.xaml">

<Application.Resources>

<Style TargetType="Button">

<Setter Property="FontFamily" Value="Comic Sans MS"/>

<Setter Property="FontSize" Value="20"/>

<Setter Property="Background" Value="#FF11C55A"/>

<Setter Property="Margin" Value="10"/>

</Style>

<Style TargetType="TextBlock">

<Setter Property="FontFamily" Value="Comic Sans MS"/>

<Setter Property="FontSize" Value="14"/>

<Setter Property="VerticalAlignment" Value="Center"/>

<Setter Property="HorizontalAlignment" Value="Center"/>

</Style>

</Application.Resources>

</Application>

* App.xaml.cs

namespace Drive

{

/// <summary>

/// Логика взаимодействия для App.xaml

/// </summary>

public partial class App : Application

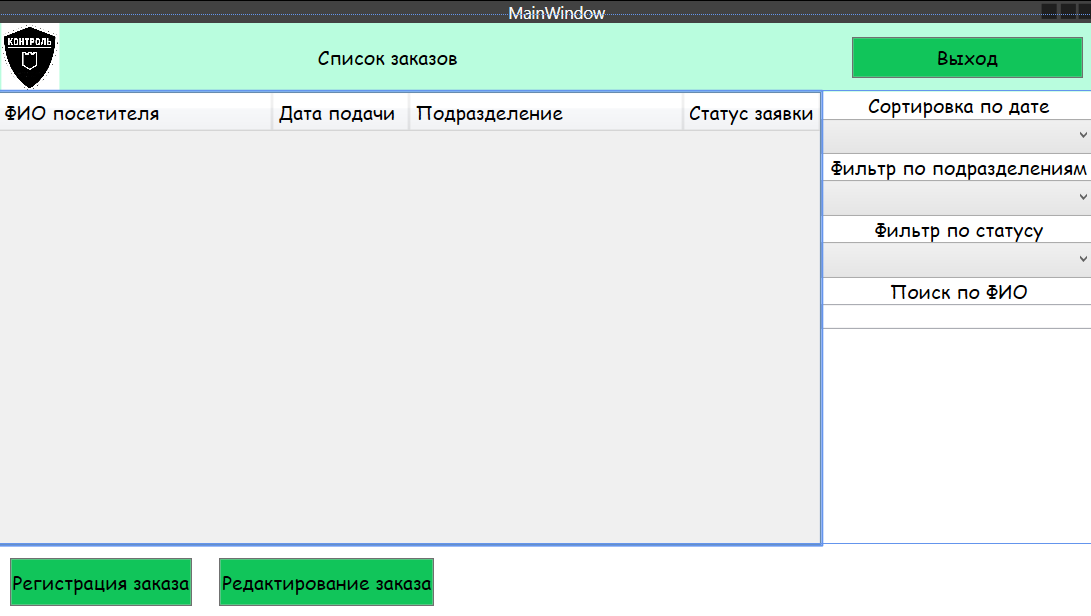
{

//объявить объект связки с бд

public static Model.DBDriveEntities DB;

}

}



Window x:Class="Drive.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:Drive"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Loaded="Window\_Loaded">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="50"/>

<RowDefinition Height="6\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<!--Заголовок-->

<Grid Grid.Row="0" Background="#FFB9FDDE">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="50"/>

<ColumnDefinition Width="95\*"/>

<ColumnDefinition Width="468\*"/>

<ColumnDefinition Width="188\*"/>

</Grid.ColumnDefinitions>

<Image Grid.Column="0" Source="/Resources/Логотип.png"/>

<TextBlock Text="Список заказов" Grid.Column="2" HorizontalAlignment="Left" Margin="89,0,0,0" />

<Button x:Name="btn\_exit" Content="Выход" Grid.Column="3" Click="btn\_exit\_Click" Margin="10,10,10,10"/>

</Grid>

<Grid Grid.Row="1">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="3\*"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<!--Таблица-->

<DataGrid Grid.Row="1" x:Name="dgRequest" AutoGenerateColumns="False" IsReadOnly="True">

<DataGrid.Columns>

<DataGridTextColumn Header="ФИО посетителя" Width="2\*" Binding="{Binding Visitor.VisitorFullName}"/>

<DataGridTextColumn Header="Дата подачи" Width="\*" Binding="{Binding RequestDateAdd, StringFormat=dd:MM:yyyy}"/>

<DataGridTextColumn Header="Подразделение" Width="2\*" Binding="{Binding Subdivision.SubdivisionName}"/>

<DataGridTextColumn Header="Статус заявки" Width="\*" Binding="{Binding Status.StatusName}"/>

</DataGrid.Columns>

</DataGrid>

<!--Фильтры и поиск-->

<StackPanel Orientation="Vertical" Grid.Column="1">

<TextBlock Text="Сортировка по дате"/>

<ComboBox x:Name="cbSort" SelectionChanged="cbSort\_SelectionChanged">

<ComboBoxItem Content="Без сортировки"/>

<ComboBoxItem Content="По возрастанию"/>

<ComboBoxItem Content="По убыванию"/>

</ComboBox>

<TextBlock Text="Фильтр по подразделениям"/>

<ComboBox x:Name="cbDivision" SelectionChanged="cbDivision\_SelectionChanged">

</ComboBox>

<TextBlock Text="Фильтр по статусу"/>

<ComboBox x:Name="cbStatus" SelectionChanged="cbStatus\_SelectionChanged">

</ComboBox>

<TextBlock Text="Поиск по ФИО"/>

<TextBox x:Name="tbFullName" TextChanged="tbFullName\_TextChanged"/>

</StackPanel>

</Grid>

<!--Кнопки управления-->

<StackPanel Orientation="Horizontal" Grid.Row="2">

<Button x:Name="btnRegistration" Content="Регистрация заказа" Click="btnRegistration\_Click" />

<Button x:Name="btnEdit" Content="Редактирование заказа" Click="btnEdit\_Click" />

</StackPanel>

</Grid>

</Window>

Код страницы Списка заявок

namespace Drive

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

//связь с бд

try

{

App.DB = new Model.DBDriveEntities();

MessageBox.Show("Подключение к бд успешно");

}

catch

{

MessageBox.Show("Связь не установлена");

App.Current.Shutdown();

}

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

ShowOrders();

cbSort.SelectedIndex = 0; //изначально выбрано

List<Model.Subdivision> divisions = App.DB.Subdivision.ToList(); //для фильтра по подразделениям

Model.Subdivision division = new Model.Subdivision(); // новое подразделение для всех подразделений сразу (как без фильтра)

division.SubdivisionID = 0;

division.SubdivisionName = "Все подразделения";

cbDivision.DisplayMemberPath = "SubdivisionName"; //видимые значения

cbDivision.SelectedValuePath = "SubdivisionID"; //получаемые значения

divisions.Insert(0, division);

cbDivision.ItemsSource = divisions;

cbDivision.SelectedIndex = 0;

List<Model.Status> statuses = App.DB.Status.ToList();

Model.Status status = new Model.Status(); // новый статус

status.StatusID = 0;

status.StatusName = "Все статусы";

cbStatus.DisplayMemberPath = "StatusName"; //видимые значения

cbStatus.SelectedValuePath = "StatusID"; //получаемые значения

cbStatus.ItemsSource = statuses;

cbStatus.SelectedIndex = 0;

statuses.Insert(0, status);

}

private void btn\_exit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

private void cbDivision\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

ShowOrders();

}

private void cbSort\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

ShowOrders();

}

public void ShowOrders()

{

//получить список всех заявок из БД

List<Model.Request> requests = App.DB.Request.ToList(); //подключение к ТАБЛИЦЕ бд

//обработка заявок

//сортировка

if (cbSort.SelectedIndex == 1)

{

requests = requests.OrderBy(r => r.RequestDateAdd /\*по какому полю сортировать заказы\*/).ToList();

}

else if (cbSort.SelectedIndex == 2)

{

requests = requests.OrderByDescending(r => r.RequestDateAdd).ToList();

}

//фильтрация по подразделениям

if (cbDivision.SelectedIndex > 0)

{

requests = requests.Where(r => r.RequestSubdivision == (int)cbDivision.SelectedValue).ToList();

}

//фильтрация по статусу

if (cbStatus.SelectedIndex > 0)

{

requests = requests.Where(r => r.RequestStatus == (int)cbStatus.SelectedValue).ToList();

}

//поиск по фио

if (tbFullName.Text.Length != 0)

{

requests = requests.Where(r => r.Visitor.VisitorFullName.ToLower().Contains(tbFullName.Text.ToLower())).ToList();

}

//отобразить список отсортированных заявок в окне

dgRequest.ItemsSource = requests;

}

private void cbStatus\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

ShowOrders();

}

private void tbFullName\_TextChanged(object sender, TextChangedEventArgs e)

{

ShowOrders();

}

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

{

View.AddOrder viewAddOrder = new View.AddOrder();

this.Hide();

viewAddOrder.ShowDialog();

this.ShowDialog();

ShowOrders(); //обновление списка грида

}

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

{

if(dgRequest.SelectedItem != null)

{

View.EditOrder viewEditOrder = new View.EditOrder((Model.Request)dgRequest.SelectedItem);

this.Hide();

viewEditOrder.ShowDialog();

this.ShowDialog();

ShowOrders(); //обновление списка грида

}

else

{

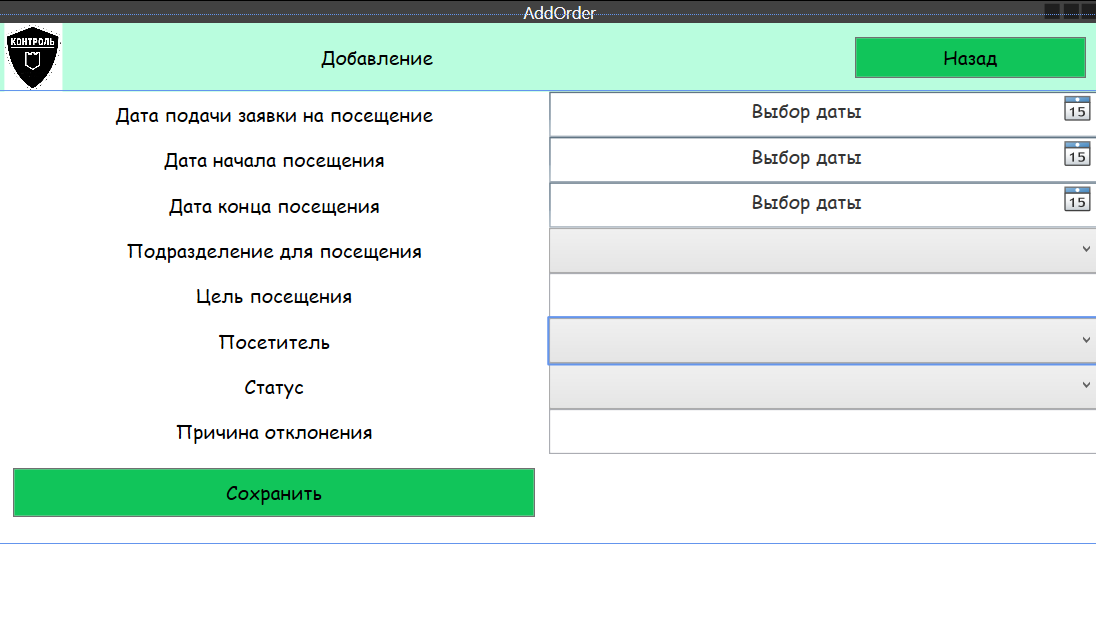
MessageBox.Show("Выберите запись");

}

}

}

}



<Window x:Class="Drive.View.AddOrder"

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:Drive.View"

mc:Ignorable="d"

Title="AddOrder" Height="450" Width="800" Loaded="Window\_Loaded">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="50"/>

<RowDefinition Height="6\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<!--Заголовок-->

<Grid Grid.Row="0" Background="#FFB9FDDE">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="50"/>

<ColumnDefinition Width="95\*"/>

<ColumnDefinition Width="468\*"/>

<ColumnDefinition Width="188\*"/>

</Grid.ColumnDefinitions>

<Image Grid.Column="0" Source="/Resources/Логотип.png"/>

<TextBlock Text="Добавление" Grid.Column="2" HorizontalAlignment="Left" Margin="89,0,0,0" />

<Button x:Name="btn\_exit" Content="Назад" Grid.Column="3" Click="btn\_exit\_Click" Margin="10,10,10,10"/>

</Grid>

<UniformGrid Grid.Row="1" Columns="2" Rows="10">

<TextBlock Text="Дата подачи заявки на посещение"/>

<DatePicker x:Name="dpDateAdd"/>

<TextBlock Text="Дата начала посещения"/>

<DatePicker x:Name="dpDateStart"/>

<TextBlock Text="Дата конца посещения"/>

<DatePicker x:Name="dpDateEnd"/>

<TextBlock Text="Подразделение для посещения"/>

<ComboBox x:Name="cbDivision">

</ComboBox>

<TextBlock Text="Цель посещения"/>

<TextBox x:Name="tbPurpose"/>

<TextBlock Text="Посетитель"/>

<ComboBox Name="cbVisitor">

</ComboBox>

<TextBlock Text="Статус"/>

<ComboBox Name="cbStatus">

</ComboBox>

<TextBlock Text="Причина отклонения"/>

<TextBox x:Name="tbReason"/>

<Button x:Name="btnSave" Content="Сохранить" Margin="10,10,10,-13" Click="btnSave\_Click"/>

</UniformGrid>

</Grid>

</Window>

Код страницы Добавления заявок

namespace Drive.View

{

/// <summary>

/// Логика взаимодействия для AddOrder.xaml

/// </summary>

public partial class AddOrder : Window

{

public AddOrder()

{

InitializeComponent();

}

private void btn\_exit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

dpDateAdd.DisplayDateStart = DateTime.Now.Date;

dpDateEnd.DisplayDateStart = DateTime.Now.Date;

dpDateStart.DisplayDateStart = DateTime.Now.Date;

List<Model.Subdivision> divisions = App.DB.Subdivision.ToList();

cbDivision.DisplayMemberPath = "SubdivisionName"; //видимые значения

cbDivision.SelectedValuePath = "SubdivisionID"; //получаемые значения

cbDivision.ItemsSource = divisions;

cbDivision.SelectedIndex = 0;

List<Model.Status> statuses = App.DB.Status.ToList();

cbStatus.DisplayMemberPath = "StatusName"; //видимые значения

cbStatus.SelectedValuePath = "StatusID"; //получаемые значения

cbStatus.ItemsSource = statuses;

cbStatus.SelectedIndex = 0;

List<Model.Visitor> visitors = App.DB.Visitor.ToList();

cbVisitor.DisplayMemberPath = "VisitorFullName"; //видимые значения

cbVisitor.SelectedValuePath = "VisitorID"; //получаемые значения

cbVisitor.ItemsSource = visitors;

cbVisitor.SelectedIndex = 0;

}

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

{

Model.Request request = new Model.Request();

request.RequestDateAdd = (DateTime)dpDateAdd.SelectedDate;

request.RequestDateStart = (DateTime)dpDateStart.SelectedDate;

request.RequestDateEnd = (DateTime)dpDateEnd.SelectedDate;

request.RequestSubdivision = (int)cbDivision.SelectedValue;

request.RequestPurpose = tbPurpose.Text;

request.RequestVisitor = (int)cbVisitor.SelectedValue;

request.RequestStatus = (int)cbStatus.SelectedValue;

App.DB.Request.Add(request);

try

{

App.DB.SaveChanges();

MessageBox.Show("Сохранено");

}

catch (Exception)

{

MessageBox.Show("Ошибка");

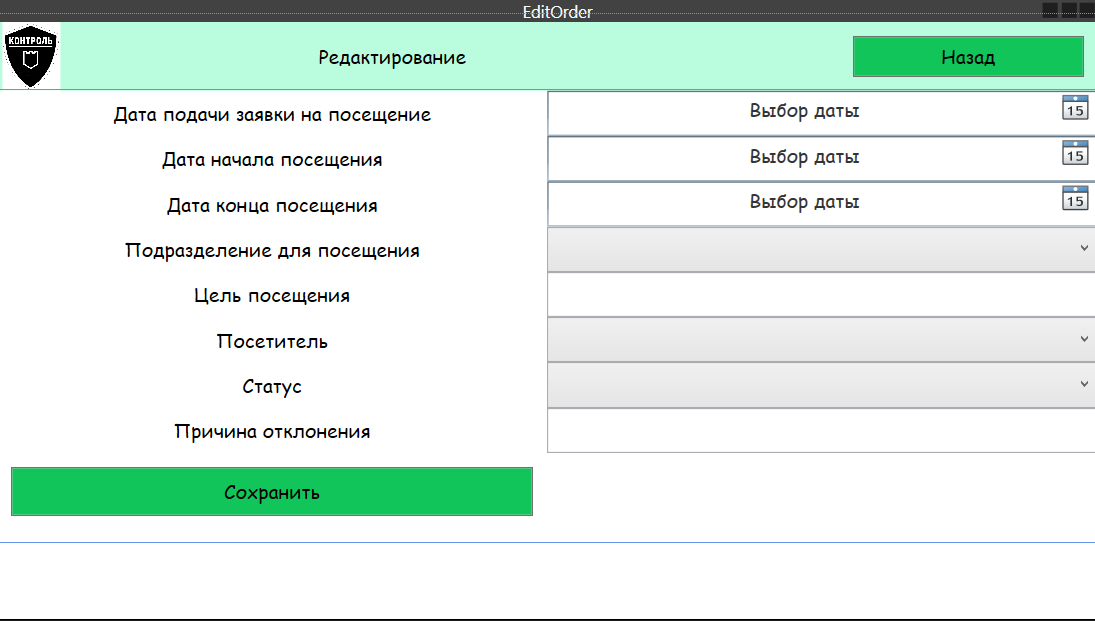
throw;

}

}

}

}



<Window x:Class="Drive.View.EditOrder"

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:Drive.View"

mc:Ignorable="d"

Title="EditOrder" Height="450" Width="800">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="50"/>

<RowDefinition Height="6\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<!--Заголовок-->

<Grid Grid.Row="0" Background="#FFB9FDDE">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="50"/>

<ColumnDefinition Width="95\*"/>

<ColumnDefinition Width="468\*"/>

<ColumnDefinition Width="188\*"/>

</Grid.ColumnDefinitions>

<Image Grid.Column="0" Source="/Resources/Логотип.png"/>

<TextBlock Text="Редактирование" Grid.Column="2" HorizontalAlignment="Left" Margin="89,0,0,0" />

<Button x:Name="btn\_exit" Content="Назад" Grid.Column="3" Click="btn\_exit\_Click" Margin="10,10,10,10"/>

</Grid>

<UniformGrid Grid.Row="1" Columns="2" Rows="10">

<TextBlock Text="Дата подачи заявки на посещение"/>

<DatePicker x:Name="dpDateAdd"/>

<TextBlock Text="Дата начала посещения"/>

<DatePicker x:Name="dpDateStart"/>

<TextBlock Text="Дата конца посещения"/>

<DatePicker x:Name="dpDateEnd"/>

<TextBlock Text="Подразделение для посещения"/>

<ComboBox x:Name="cbDivision">

</ComboBox>

<TextBlock Text="Цель посещения"/>

<TextBox x:Name="tbPurpose"/>

<TextBlock Text="Посетитель"/>

<ComboBox Name="cbVisitor">

</ComboBox>

<TextBlock Text="Статус"/>

<ComboBox Name="cbStatus">

</ComboBox>

<TextBlock Text="Причина отклонения"/>

<TextBox x:Name="tbReason"/>

<Button x:Name="btnSave" Content="Сохранить" Margin="10,10,10,-13" Click="btnSave\_Click"/>

</UniformGrid>

</Grid>

</Window>

Код страницы Изменения

namespace Drive.View

{

/// <summary>

/// Логика взаимодействия для EditOrder.xaml

/// </summary>

public partial class EditOrder : Window

{

private Model.Request workRequest; //создаем объект выбранной заявки с прошлого окна

public EditOrder(Model.Request editRequest)

{

InitializeComponent();

workRequest = editRequest;

dpDateAdd.DisplayDateStart = DateTime.Now.Date;

dpDateEnd.DisplayDateStart = DateTime.Now.Date;

dpDateStart.DisplayDateStart = DateTime.Now.Date;

List<Model.Subdivision> divisions = App.DB.Subdivision.ToList();

cbDivision.DisplayMemberPath = "SubdivisionName"; //видимые значения

cbDivision.SelectedValuePath = "SubdivisionID"; //получаемые значения

cbDivision.ItemsSource = divisions;

cbDivision.SelectedIndex = 0;

List<Model.Status> statuses = App.DB.Status.ToList();

cbStatus.DisplayMemberPath = "StatusName"; //видимые значения

cbStatus.SelectedValuePath = "StatusID"; //получаемые значения

cbStatus.ItemsSource = statuses;

cbStatus.SelectedIndex = 0;

List<Model.Visitor> visitors = App.DB.Visitor.ToList();

cbVisitor.DisplayMemberPath = "VisitorFullName"; //видимые значения

cbVisitor.SelectedValuePath = "VisitorID"; //получаемые значения

cbVisitor.ItemsSource = visitors;

cbVisitor.SelectedIndex = 0;

dpDateAdd.SelectedDate = workRequest.RequestDateAdd;

dpDateStart.SelectedDate = workRequest.RequestDateStart;

dpDateEnd.SelectedDate = workRequest.RequestDateEnd;

cbDivision.SelectedValue = workRequest.RequestSubdivision;

tbPurpose.Text = workRequest.RequestPurpose;

cbVisitor.SelectedValue = workRequest.RequestVisitor;

cbStatus.SelectedValue = workRequest.RequestStatus;

}

private void btn\_exit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

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

{

workRequest.RequestDateAdd = (DateTime)dpDateAdd.SelectedDate;

workRequest.RequestDateStart = (DateTime)dpDateStart.SelectedDate;

workRequest.RequestDateEnd = (DateTime)dpDateEnd.SelectedDate;

workRequest.RequestSubdivision = (int)cbDivision.SelectedValue;

workRequest.RequestPurpose = tbPurpose.Text;

workRequest.RequestVisitor = (int)cbVisitor.SelectedValue;

workRequest.RequestStatus = (int)cbStatus.SelectedValue;

try

{

App.DB.SaveChanges();

MessageBox.Show("Сохранено");

}

catch (Exception)

{

MessageBox.Show("Ошибка");

throw;

}

}

}

}