**ПРИЛОЖЕНИЕ Д**

**Листинг кода**

using DatabaseManager;using LiveChartsCore.Kernel.Sketches;using LiveChartsCore.SkiaSharpView.Painting;using LiveChartsCore.SkiaSharpView.SKCharts;using LiveChartsCore.SkiaSharpView;using LiveChartsCore;using MaterialDesignThemes.Wpf;using Microsoft.EntityFrameworkCore;using Microsoft.Win32;using SkiaSharp;using System.Collections.Generic;using System.Collections.ObjectModel;using System.ComponentModel.DataAnnotations.Schema;using System.ComponentModel.DataAnnotations;using System.ComponentModel;using System.Diagnostics;using System.Globalization;using System.IO;using System.Runtime.CompilerServices;using System.Security.Cryptography;using System.Text.RegularExpressions;using System.Text;using System.Timers;using System.Windows.Controls;using System.Windows.Data;using System.Windows.Documents;using System.Windows.Input;using System.Windows.Media;using System.Windows;using VacTrack.DialogWindows;using VacTrack.ViewReport;namespace DatabaseManager{public class BaseModel : INotifyPropertyChanged{public event PropertyChangedEventHandler? PropertyChanged;protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));protected virtual bool SetProperty<T>(ref T member,T value,[CallerMemberName] string? propertyName=null){if(EqualityComparer<T>.Default.Equals(member,value)){return false;}member=value;OnPropertyChanged(propertyName);return true;}[NotMapped] private bool \_isSelected;[NotMapped] public bool IsSelected{get=>\_isSelected;set=>SetProperty(ref \_isSelected,value);}}}namespace DatabaseManager{public class DatabaseContext : DbContext{public DbSet<Unit>Units{get;set;}=null!;public DbSet<Post>Posts{get;set;}=null!;public DbSet<Sale>Sales{get;set;}=null!;public DbSet<Product>Products{get;set;}=null!;public DbSet<Receipt>Receipts{get;set;}=null!;public DbSet<Contract>Contracts{get;set;}=null!;public DbSet<Division>Divisions{get;set;}=null!;public DbSet<Employee>Employees{get;set;}=null!;public DbSet<Location>Locations{get;set;}=null!;public DbSet<Material>Materials{get;set;}=null!;public DbSet<Counterpartie>Counterparties{get;set;}=null!;public DbSet<Product\_Material>Product\_Materials{get;set;}=null!;public DbSet<Users>Users{get;set;}static DatabaseContext(){SQLitePCL.raw.SetProvider(new SQLitePCL.SQLite3Provider\_e\_sqlcipher());}protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder){optionsBuilder.UseSqlite("Data Source=VacDB.db;Password=MyPass12345");}protected override void OnModelCreating(ModelBuilder modelBuilder){modelBuilder.Entity<Employee>().HasOne(e=>e.Division).WithMany(d=>d.Employees).HasForeignKey(e=>e.DivisionId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Employee>().HasOne(e=>e.Post).WithMany(p=>p.Employees).HasForeignKey(e=>e.PostId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Counterpartie>(entity=>{entity.Property(c=>c.Type).HasConversion<string>().HasDefaultValue(CounterpartieType.Fiz).HasColumnType("TEXT");entity.ToTable(t=>t.HasCheckConstraint("CK\_Customer\_Type","Type IN ('Fiz', 'Ur')"));});modelBuilder.Entity<Location>().HasOne(l=>l.Employee).WithMany().HasForeignKey(l=>l.EmployeeId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Product>().HasOne(p=>p.Unit).WithMany(u=>u.Products).HasForeignKey(p=>p.UnitId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Material>().HasOne(m=>m.Unit).WithMany(u=>u.Materials).HasForeignKey(m=>m.UnitId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Product>().HasOne(p=>p.Location).WithMany(l=>l.Products).HasForeignKey(p=>p.LocationId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Material>().HasOne(m=>m.Location).WithMany(l=>l.Materials).HasForeignKey(m=>m.LocationId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Product\_Material>().HasKey(pm=>new{pm.Id});modelBuilder.Entity<Product\_Material>().HasOne(pm=>pm.Product).WithMany(p=>p.ProductMaterials).HasForeignKey(pm=>pm.ProductId);modelBuilder.Entity<Product\_Material>().HasOne(pm=>pm.Material).WithMany(m=>m.ProductMaterials).HasForeignKey(pm=>pm.MaterialId);modelBuilder.Entity<Contract>().HasOne(c=>c.Counterpartie).WithMany(cp=>cp.Contracts).HasForeignKey(c=>c.CounterpartyId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Receipt>().HasOne(r=>r.Counterpartie).WithMany(cp=>cp.Receipts).HasForeignKey(r=>r.CounterpartyId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Contract>().HasOne(c=>c.Product).WithMany(p=>p.Contracts).HasForeignKey(c=>c.ProductId).OnDelete(DeleteBehavior.NoAction);modelBuilder.Entity<Sale>().HasOne(s=>s.Contract).WithMany(c=>c.Sales).HasForeignKey(s=>s.ContractId).OnDelete(DeleteBehavior.NoAction);}}}namespace DatabaseManager{public enum ContractStatus{created,running,completed}public class Contract : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private int \_counterpartyId;public int CounterpartyId{get=>\_counterpartyId;set{\_counterpartyId=value;OnPropertyChanged(nameof(CounterpartyId));}}private Counterpartie? \_counterpartie;public Counterpartie? Counterpartie{get=>\_counterpartie;set{\_counterpartie=value;OnPropertyChanged(nameof(Counterpartie));}}private DateTime \_date;public DateTime Date{get=>\_date;set{\_date=value;OnPropertyChanged(nameof(Date));}}private int \_term;public int Term{get=>\_term;set{\_term=value;OnPropertyChanged(nameof(Term));}}private double \_summ;public double Summ{get=>Math.Round(\_summ,2);set{\_summ=value;OnPropertyChanged(nameof(Summ));}}private int \_productId;public int ProductId{get=>\_productId;set{\_productId=value;OnPropertyChanged(nameof(ProductId));}}private Product? \_product;public Product? Product{get=>\_product;set{\_product=value;OnPropertyChanged(nameof(Product));}}private int \_count;public int Count{get=>\_count;set{\_count=value>0 ? value : 0;OnPropertyChanged(nameof(Count));}}private List<Sale>\_sales=[];public List<Sale>Sales{get=>\_sales;set{\_sales=value;OnPropertyChanged(nameof(Sales));}}private ContractStatus \_status=ContractStatus.created;public ContractStatus Status{get=>\_status;set=>SetProperty(ref \_status,value);}[System.ComponentModel.DataAnnotations.Schema.NotMapped] public string DisplayFull=>$"Назвение: {Name};\nКому: {Counterpartie?.Name};\nИзделие: {Product?.Name}";}}namespace DatabaseManager{public enum CounterpartieType{Fiz,Ur}public class Counterpartie : BaseModel{private int \_Id;public int Id{get=>\_Id;set=>SetProperty(ref \_Id,value);}private string \_Name;public required string Name{get=>\_Name;set=>SetProperty(ref \_Name,value);}private string \_LegalAddress;public required string LegalAddress{get=>\_LegalAddress;set=>SetProperty(ref \_LegalAddress,value);}private string \_PhoneNomber;public required string PhoneNomber{get=>\_PhoneNomber;set=>SetProperty(ref \_PhoneNomber,value);}private string \_PostalAddress;public required string PostalAddress{get=>\_PostalAddress;set=>SetProperty(ref \_PostalAddress,value);}private string \_Unp;public required string Unp{get=>\_Unp;set=>SetProperty(ref \_Unp,value);}private string \_bankAccount;public required string BankAccount{get=>\_bankAccount;set=>SetProperty(ref \_bankAccount,value);}private CounterpartieType \_Type=CounterpartieType.Fiz;public CounterpartieType Type{get=>\_Type;set=>SetProperty(ref \_Type,value);}private string? \_Okulp;public string? Okulp{get=>\_Okulp;set=>SetProperty(ref \_Okulp,value);}private string? \_Okpo;public string? Okpo{get=>\_Okpo;set=>SetProperty(ref \_Okpo,value);}private string? \_Oked;public string? Oked{get=>\_Oked;set=>SetProperty(ref \_Oked,value);}private List<Contract>\_Contracts=[];public List<Contract>Contracts{get=>\_Contracts;set=>SetProperty(ref \_Contracts,value);}private List<Receipt>\_Receipts=[];public List<Receipt>Receipts{get=>\_Receipts;set=>SetProperty(ref \_Receipts,value);}}}namespace DatabaseManager{public class Division : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private List<Employee>\_employees=[];public List<Employee>Employees{get=>\_employees;set{\_employees=value;OnPropertyChanged(nameof(Employees));}}}}namespace DatabaseManager{public class Employee : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_fio;public required string Fio{get=>\_fio ?? throw new InvalidOperationException("Fio must be initialized.");set{\_fio=value;OnPropertyChanged(nameof(Fio));}}private int \_divisionId;public int DivisionId{get=>\_divisionId;set{\_divisionId=value;OnPropertyChanged(nameof(DivisionId));}}private Division? \_division;public Division? Division{get=>\_division;set{\_division=value;OnPropertyChanged(nameof(Division));}}private int \_postId;public int PostId{get=>\_postId;set{\_postId=value;OnPropertyChanged(nameof(PostId));}}private Post? \_post;public Post? Post{get=>\_post;set{\_post=value;OnPropertyChanged(nameof(Post));}}private DateTime \_dateHire;public DateTime DateHire{get=>\_dateHire;set{\_dateHire=value;OnPropertyChanged(nameof(DateHire));}}private DateTime? \_dateDismissal;public DateTime? DateDismissal{get=>\_dateDismissal;set{\_dateDismissal=value;OnPropertyChanged(nameof(DateDismissal));}}private DateTime \_dateOfBirth;public DateTime DateOfBirth{get=>\_dateOfBirth;set{\_dateOfBirth=value;OnPropertyChanged(nameof(DateOfBirth));}}[StringLength(13,MinimumLength=10,ErrorMessage="The line length must be exactly 10-13 characters.")] private string? \_phoneNumber;public required string PhoneNumber{get=>\_phoneNumber ?? throw new InvalidOperationException("PhoneNumber must be initialized.");set{\_phoneNumber=value;OnPropertyChanged(nameof(PhoneNumber));}}private string? \_address;public required string Address{get=>\_address ?? throw new InvalidOperationException("Address must be initialized.");set{\_address=value;OnPropertyChanged(nameof(Address));}}private string? \_passportData;public required string PassportData{get=>\_passportData ?? throw new InvalidOperationException("PassportData must be initialized.");set{\_passportData=value;OnPropertyChanged(nameof(PassportData));}}private double \_salary;public double Salary{get=>Math.Round(\_salary,2);set{\_salary=value;OnPropertyChanged(nameof(Salary));}}private double? \_bonuses;public double? Bonuses{get=>Math.Round(\_bonuses ?? 0,3);set{\_bonuses=value;OnPropertyChanged(nameof(Bonuses));}}private string? \_bankDetails;public required string BankDetails{get=>\_bankDetails ?? throw new InvalidOperationException("BankDetails must be initialized.");set{\_bankDetails=value;OnPropertyChanged(nameof(BankDetails));}}[System.ComponentModel.DataAnnotations.Schema.NotMapped] public string GetFioAndPost{get=>$"{Fio}\t({Post?.Name})";}}}namespace DatabaseManager{public class Location : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private int \_employeeId;public int EmployeeId{get=>\_employeeId;set{\_employeeId=value;OnPropertyChanged(nameof(EmployeeId));}}private Employee? \_employee;public Employee? Employee{get=>\_employee;set{\_employee=value;OnPropertyChanged(nameof(Employee));}}private List<Product>\_products=new List<Product>();public List<Product>Products{get=>\_products;set{\_products=value;OnPropertyChanged(nameof(Products));}}private List<Material>\_materials=new List<Material>();public List<Material>Materials{get=>\_materials;set{\_materials=value;OnPropertyChanged(nameof(Materials));}}}}namespace DatabaseManager{public class Material : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private int \_unitId;public int UnitId{get=>\_unitId;set{\_unitId=value;OnPropertyChanged(nameof(UnitId));}}private Unit? \_unit;public Unit? Unit{get=>\_unit;set{\_unit=value;OnPropertyChanged(nameof(Unit));}}private int \_locationId;public int LocationId{get=>\_locationId;set{\_locationId=value;OnPropertyChanged(nameof(LocationId));}}private Location? \_location;public Location? Location{get=>\_location;set{\_location=value;OnPropertyChanged(nameof(Location));}}private double \_count;public double Count{get=>Math.Round(\_count,3);set{\_count=value>0 ? value : 0;OnPropertyChanged(nameof(Count));OnPropertyChanged(nameof(GetSum));}}private double \_price;public double Price{get=>Math.Round(\_price,2);set{\_price=value>0 ? value : 0;OnPropertyChanged(nameof(Price));OnPropertyChanged(nameof(GetSum));}}private List<Product\_Material>\_productMaterials=[];public List<Product\_Material>ProductMaterials{get=>\_productMaterials;set{\_productMaterials=value;OnPropertyChanged(nameof(ProductMaterials));}}private List<Receipt>\_receipts=[];public List<Receipt>Receipts{get=>\_receipts;set{\_receipts=value;OnPropertyChanged(nameof(Receipts));}}[System.ComponentModel.DataAnnotations.Schema.NotMapped] public double GetSum{get=>Math.Round(Count\*Price,3);}}}namespace DatabaseManager{public class Post : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private List<Employee>\_employees=new List<Employee>();public List<Employee>Employees{get=>\_employees;set{\_employees=value;OnPropertyChanged(nameof(Employees));}}}}namespace DatabaseManager{public class Product : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private string? \_serialNo;public required string SerialNo{get=>\_serialNo ?? throw new InvalidOperationException("SerialNo must be initialized.");set{\_serialNo=value;OnPropertyChanged(nameof(SerialNo));}}private int \_unitId;public int UnitId{get=>\_unitId;set{\_unitId=value;OnPropertyChanged(nameof(UnitId));}}private Unit? \_unit;public Unit? Unit{get=>\_unit;set{\_unit=value;OnPropertyChanged(nameof(Unit));}}private int \_locationId;public int LocationId{get=>\_locationId;set{\_locationId=value;OnPropertyChanged(nameof(LocationId));}}private Location? \_location;public Location? Location{get=>\_location;set{\_location=value;OnPropertyChanged(nameof(Location));}}private List<Product\_Material>\_productMaterials=[];public List<Product\_Material>ProductMaterials{get=>\_productMaterials;set{\_productMaterials=value;OnPropertyChanged(nameof(ProductMaterials));}}private List<Contract>\_contracts=[];public List<Contract>Contracts{get=>\_contracts;set{\_contracts=value;OnPropertyChanged(nameof(Contracts));}}private int \_count;public int Count{get=>\_count;set{\_count=value>0 ? value : 0;OnPropertyChanged(nameof(Count));}}private double \_price;public double Price{get=>Math.Round(Math.Round(\_price,2));set{\_price=Math.Round(value>0 ? value : 0,3);OnPropertyChanged(nameof(Price));OnPropertyChanged(nameof(SummNds));OnPropertyChanged(nameof(Summ));}}[NotMapped] public double Nds;[NotMapped] public double SummNds{get=>Math.Round(Price\*(Nds/100),3);}[NotMapped] public double Summ{get=>Math.Round(Price+SummNds,3);}}}namespace DatabaseManager{public class Product\_Material : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private int \_productId;public int ProductId{get=>\_productId;set{\_productId=value;OnPropertyChanged(nameof(ProductId));}}private Product? \_product;public Product? Product{get=>\_product;set{\_product=value;OnPropertyChanged(nameof(Product));}}private int \_materialId;public int MaterialId{get=>\_materialId;set{\_materialId=value;OnPropertyChanged(nameof(MaterialId));}}private Material? \_material;public Material? Material{get=>\_material;set{\_material=value;OnPropertyChanged(nameof(Material));}}private int \_quantity;public int Quantity{get=>\_quantity;set{\_quantity=value>0 ? value : 0;OnPropertyChanged(nameof(Quantity));}}[System.ComponentModel.DataAnnotations.Schema.NotMapped] public double GetSum{get=>Material==null ? 0 : Quantity\*Material.Price;}}}namespace DatabaseManager{public class Receipt : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private int \_materialId;public int MaterialId{get=>\_materialId;set{\_materialId=value;OnPropertyChanged(nameof(MaterialId));}}private Material? \_material;public Material? Material{get=>\_material;set{\_material=value;OnPropertyChanged(nameof(Material));Summ=Count\*(Material!=null ? Material.Price : 0);}}private double \_summ;public double Summ{get=>Math.Round(\_summ,2);set{\_summ=value;OnPropertyChanged(nameof(Summ));}}private DateTime \_date;public DateTime Date{get=>\_date;set{\_date=value;OnPropertyChanged(nameof(Date));}}private int \_count;public int Count{get=>\_count;set{\_count=value;OnPropertyChanged(nameof(Count));Summ=Count\*(Material!=null ? Material.Price : 0);}}private int \_counterpartyId;public int CounterpartyId{get=>\_counterpartyId;set{\_counterpartyId=value;OnPropertyChanged(nameof(CounterpartyId));}}private Counterpartie? \_counterpartie;public Counterpartie? Counterpartie{get=>\_counterpartie;set{\_counterpartie=value;OnPropertyChanged(nameof(Counterpartie));}}}}namespace DatabaseManager{public class Sale : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private int \_contractId;public int ContractId{get=>\_contractId;set{\_contractId=value;OnPropertyChanged(nameof(ContractId));}}private Contract? \_contract;public Contract? Contract{get=>\_contract;set{\_contract=value;OnPropertyChanged(nameof(Contract));}}private double \_summ;public double Summ{get=>Math.Round(\_summ,2);set{\_summ=value;OnPropertyChanged(nameof(Summ));}}private DateTime \_date;public DateTime Date{get=>\_date;set{\_date=value;OnPropertyChanged(nameof(Date));}}private int \_count;public int Count{get=>\_count;set{\_count=value>0 ? value : 0;OnPropertyChanged(nameof(Count));}}}}namespace DatabaseManager{public class Unit : BaseModel{private int \_id;public int Id{get=>\_id;set{\_id=value;OnPropertyChanged(nameof(Id));}}private string? \_name;public required string Name{get=>\_name ?? throw new InvalidOperationException("Name must be initialized.");set{\_name=value;OnPropertyChanged(nameof(Name));}}private List<Product>\_products=[];public List<Product>Products{get=>\_products;set{\_products=value;OnPropertyChanged(nameof(Products));}}private List<Material>\_materials=[];public List<Material>Materials{get=>\_materials;set{\_materials=value;OnPropertyChanged(nameof(Materials));}}private string? \_Description;public string? Description{get=>\_Description;set{\_Description=value;OnPropertyChanged(nameof(Description));}}}}namespace DatabaseManager{public class Users : BaseModel{private int \_id;public int Id{get=>\_id;set{SetProperty(ref \_id,value);}}private string? \_login;public required string Login{get=>\_login ?? throw new InvalidOperationException("Login must be initialized.");set{SetProperty(ref \_login,value);}}private string? \_pass;public required string Password{get=>\_pass ?? throw new InvalidOperationException("Password must be initialized.");set{SetProperty(ref \_pass,value);}}private string? \_access;public string? Access{get=>\_access;set{SetProperty(ref \_access,value);}}}}<Window x:Class="VacTrack.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:VacTrack" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" mc:Ignorable="d" Title="«Учёт выпуска и реализации продукции в ООО «Вактайм» г. Сморгонь»" Height="500" Width="900" MinHeight="450" MinWidth="800"><Window.DataContext><local:MainWindowViewModel/></Window.DataContext><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><Menu IsMainMenu="True" VerticalAlignment="Top" materialDesign:MenuAssist.TopLevelMenuItemHeight="20"><MenuItem Header="Файл"><MenuItem Header="Настройки" Click="NavigateToPage" Tag="Settings"/><MenuItem Header="Создать новое окно" Click="CreateNewWindow"/><MenuItem Header="Начальний экран" Click="NavigateToPage" Tag="Home"/><MenuItem Header="Выход" Click="LogOunt"/></MenuItem><MenuItem Header="Справочники" Visibility="{Binding IsLoginVisable}"><MenuItem Header="Сотрудники" Click="NavigateToPage" Tag="Employee"/><MenuItem Header="Должности" Click="NavigateToPage" Tag="Post"/><MenuItem Header="Продукция" Click="NavigateToPage" Tag="Product"/><MenuItem Header="Единицы измерения" Click="NavigateToPage" Tag="Unit"/><MenuItem Header="Места хранения" Click="NavigateToPage" Tag="Location"/><MenuItem Header="Подразделения" Click="NavigateToPage" Tag="Division"/><MenuItem Header="Материалы\Коплектуюшие" Click="NavigateToPage" Tag="Material"/><MenuItem Header="Контрагенты" Click="NavigateToPage" Tag="Counterpartie"/></MenuItem><MenuItem Header="Документы" Visibility="{Binding IsLoginVisable}"><MenuItem Header="Договора" Click="NavigateToPage" Tag="Contract"/><MenuItem Header="Поступления" Click="NavigateToPage" Tag="Receipt"/><MenuItem Header="Реализация" Click="NavigateToPage" Tag="Sale"/></MenuItem><MenuItem Header="Отчёты" Visibility="{Binding IsLoginVisable}"><MenuItem Header="Отчет по использованию материалов на изделия" Click="NavigateToPage" Tag="MaterialUsageReport" ToolTip="Показывает, какие материалы использовались для производства определенных изделий, их количество и стоимость."/><MenuItem Header="Отчет по сотрудникам" Click="NavigateToPage" Tag="EmployeeDivisionReport" ToolTip="Список всех сотрудников с указанием их подразделений, должностей, даты приема и даты увольнения (если есть)."/><MenuItem Header="Отчет по остаткам материалов и комплектуюших" Click="NavigateToPage" Tag="StockBalanceReport" ToolTip="Информация о текущих остатках материалов и комплектуюших с указанием мест хранения."/><MenuItem Header="Отчет по договорам с контрагентами" Click="NavigateToPage" Tag="ContractorContractsReport" ToolTip="Информация о заключенных договорах с контрагентами, их суммах и изделиях."/><MenuItem Header="Отчет по реализации продукции" Click="NavigateToPage" Tag="ProductSalesReport" ToolTip="Детализация продаж продукции: что было продано, кому, за какую сумму и в каком количестве."/><MenuItem Header="Лист закупок" Click="NavigateToPage" Tag="ProcurementSheetReport" ToolTip="Формирование списка закупок"/><MenuItem Header="Статистика продаж" Click="NavigateToPage" Tag="SalesStatistics"/></MenuItem><MenuItem Header="Справка"><MenuItem Header="О программе" Click="OpenAboutProgram"/><MenuItem Header="Помощь" Click="OpenHelp"/></MenuItem><MenuItem Header="Создать пользователя" Click="CreateNewUser\_click" Visibility="{Binding NegativeIsLoginVisable}"/></Menu><StackPanel Orientation="Vertical" Visibility="{Binding NegativeIsLoginVisable}" VerticalAlignment="Center" HorizontalAlignment="Center" Margin="0,10,0,0"><Label Content="Вход" HorizontalAlignment="Center" FontSize="20"/><TextBox Style="{StaticResource MaterialDesignOutlinedTextBox}" Width="300" materialDesign:HintAssist.Hint="Логин" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding LoginString}" Margin="0,10,0,0"/><PasswordBox materialDesign:HintAssist.Hint="Пароль" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" local:PasswordBoxHelper.BoundPassword="{Binding PasswordString, Mode=TwoWay, UpdateSourceTrigger=PropertyChanged}" Margin="0,10,0,0"/><CheckBox Content="Запомнить" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,10,0,0" IsChecked="{Binding SaveUser}"/><Label Content="{Binding Message}" Foreground="#E5FFDE00" Height="25" HorizontalAlignment="Center"/><Button Content="Вход" Style="{StaticResource MaterialDesignPaperButton}" Command="{Binding LogInCommand}" Cursor="Hand" Margin="0,10,0,0"/></StackPanel><Frame x:Name="MainFrame" NavigationUIVisibility="Hidden" Margin="5,25,5,5" Visibility="{Binding IsLoginVisable}"/></Grid></Window>namespace VacTrack{public partial class MainWindow : Window{private readonly Dictionary<string,Page>\_pagesCache=[];private readonly PaletteHelper \_paletteHelper=new();private MainWindowViewModel ThisViewModel=>(MainWindowViewModel)DataContext;public MainWindow(){var Db=new DatabaseContext();Db.Database.EnsureCreated();InitializeComponent();MainFrame.Navigate(new HomePage());SetTheme(Properties.Settings.Default.AppTheme);}private void SetTheme(string SelectTheme){var theme=\_paletteHelper.GetTheme();switch(SelectTheme){case "Dark": theme.SetBaseTheme(BaseTheme.Dark);break;case "Light": theme.SetBaseTheme(BaseTheme.Light);break;default:{theme.SetBaseTheme(BaseTheme.Inherit);}break;}\_paletteHelper.SetTheme(theme);}private void Close(object sender,RoutedEventArgs e)=>Close();private void LogOunt(object sender,RoutedEventArgs e){Properties.Settings.Default.LogInUserId=-1;ThisViewModel.IsLogin=false;Properties.Settings.Default.Save();}private void OpenAboutProgram(object sender,RoutedEventArgs e)=>new AboutProgram().ShowDialog();private void CreateNewUser\_click(object sender,RoutedEventArgs e)=>new CreateUser().ShowDialog();private void CreateNewWindow(object sender,RoutedEventArgs e)=>new MainWindow().Show();private void OpenHelp(object sender,RoutedEventArgs e)=>OpenChmFile(@"UserGuide.chm");static void OpenChmFile(string filePath){if(string.IsNullOrEmpty(filePath)){MessageBox.Show("Путь к файлу справки не указан.");return;}if(!System.IO.File.Exists(filePath)){MessageBox.Show($"Файл справки не найден: {filePath}");return;}try{System.Diagnostics.Process.Start(new System.Diagnostics.ProcessStartInfo{FileName="hh.exe",Arguments=filePath,UseShellExecute=true});}catch(Exception ex){System.Diagnostics.Debug.WriteLine($"Не удалось открыть файл: {ex.Message}");MessageBox.Show($"Не удалось открыть файл: {ex.Message}");}}private void NavigateToPage(object sender,RoutedEventArgs e){try{var menuItem=sender as MenuItem;string? pageKey=menuItem?.Tag.ToString();if(string.IsNullOrEmpty(pageKey))return;if(!\_pagesCache.TryGetValue(pageKey,out Page? targetPage)){targetPage=pageKey switch{"Contract"=>new ViewTables.ContractViewTable(),"Counterpartie"=>new ViewTables.CounterpartieViewTable(),"Division"=>new ViewTables.DivisionViewTable(),"Employee"=>new ViewTables.EmployeeViewTable(),"Location"=>new ViewTables.LocationViewTable(),"Material"=>new ViewTables.MaterialViewTable(),"Post"=>new ViewTables.PostViewTable(),"Product"=>new ViewTables.ProductViewTable(),"Receipt"=>new ViewTables.ReceiptViewTable(),"Sale"=>new ViewTables.SaleViewTable(),"Unit"=>new ViewTables.UnitViewTable(),"Home"=>new HomePage(),"MaterialUsageReport"=>new ViewReport.MaterialUsageReport(),"EmployeeDivisionReport"=>new ViewReport.EmployeeDivisionReport(),"StockBalanceReport"=>new ViewReport.StockBalanceReport(),"ContractorContractsReport"=>new ViewReport.ContractorContractsReport(),"ProductSalesReport"=>new ViewReport.ProductSalesReport(),"ProcurementSheetReport"=>new ViewReport.ProcurementSheetReport(),"SalesStatistics"=>new ViewReport.SalesStatisticsReport(),"Settings"=>new SettingsPage(),\_=>new NotFoundPage("Запрашиваемая страница не найдена"),};\_pagesCache[pageKey]=targetPage;}else{if(targetPage is ICachedPage cachedPage)cachedPage.OnNavigatedFromCache();}MainFrame.Navigate(targetPage);}catch(Exception ex){MessageBox.Show(ex.Message);}}}public class MainWindowViewModel : INotifyPropertyChanged{#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion private bool \_isLogin=false;public bool IsLogin{get=>\_isLogin;set{SetProperty(ref \_isLogin,value);OnPropertyChanged(nameof(IsLoginVisable));OnPropertyChanged(nameof(NegativeIsLoginVisable));}}private string \_message=string.Empty;public string Message{get=>\_message;set{SetProperty(ref \_message,value);}}public Visibility IsLoginVisable=>IsLogin ? Visibility.Visible : Visibility.Collapsed;public Visibility NegativeIsLoginVisable=>!IsLogin ? Visibility.Visible : Visibility.Collapsed;public bool SaveUser{get;set;}=false;public string? LoginString{get;set;}public string? PasswordString{get;set;}private List<Users>Users{get;set;}=[];public ICommand LogInCommand{get;set;}public MainWindowViewModel(){LogInCommand=new RelayCommand(LogIn);if(Properties.Settings.Default.LogInUserId!=-1)IsLogin=true;}private void LogIn(object obj){var Db=new DatabaseContext();Users=[..Db.Users];if(string.IsNullOrWhiteSpace(PasswordString)||string.IsNullOrWhiteSpace(LoginString)){Message="Введите пароль и логин";return;}var user=Users.FirstOrDefault(u=>u.Login==LoginString);IsLogin=user!=null&&user.Password==CreateUserViewModel.HashPassword(PasswordString);if(SaveUser){Properties.Settings.Default.LogInUserId=IsLogin ? user?.Id ??-1 :-1;Properties.Settings.Default.Save();}Message=IsLogin ? "Успешно" : "Не верный логин или пароль";}}}using System.Windows;namespace VacTrack{public static class PasswordBoxHelper{public static readonly DependencyProperty BoundPasswordProperty=DependencyProperty.RegisterAttached("BoundPassword",typeof(string),typeof(PasswordBoxHelper),new PropertyMetadata(string.Empty,OnBoundPasswordChanged));public static string GetBoundPassword(DependencyObject obj)=>(string)obj.GetValue(BoundPasswordProperty);public static void SetBoundPassword(DependencyObject obj,string value)=>obj.SetValue(BoundPasswordProperty,value);private static void OnBoundPasswordChanged(DependencyObject d,DependencyPropertyChangedEventArgs e){if(d is PasswordBox passwordBox){passwordBox.PasswordChanged-=PasswordBox\_PasswordChanged;if(!GetIsUpdating(passwordBox))passwordBox.Password=(string)e.NewValue;passwordBox.PasswordChanged+=PasswordBox\_PasswordChanged;}}private static readonly DependencyProperty IsUpdatingProperty=DependencyProperty.RegisterAttached("IsUpdating",typeof(bool),typeof(PasswordBoxHelper));private static bool GetIsUpdating(DependencyObject obj)=>(bool)obj.GetValue(IsUpdatingProperty);private static void SetIsUpdating(DependencyObject obj,bool value)=>obj.SetValue(IsUpdatingProperty,value);private static void PasswordBox\_PasswordChanged(object sender,RoutedEventArgs e){if(sender is PasswordBox passwordBox){SetIsUpdating(passwordBox,true);SetBoundPassword(passwordBox,passwordBox.Password);SetIsUpdating(passwordBox,false);}}}}public class RelayCommand : ICommand{private readonly Action<object>\_execute;private readonly Predicate<object>\_canExecute;public RelayCommand(Action<object>execute,Predicate<object>canExecute=null){\_execute=execute;\_canExecute=canExecute;}public bool CanExecute(object parameter)=>\_canExecute?.Invoke(parameter)?? true;public void Execute(object parameter)=>\_execute(parameter);public event EventHandler CanExecuteChanged{add=>CommandManager.RequerySuggested+=value;remove=>CommandManager.RequerySuggested-=value;}}<Page x:Class="VacTrack.SettingsPage" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="SettingsPage"><Page.DataContext><local:SettingViewModel/></Page.DataContext><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><TabControl Margin="0,0,0,35"><TabItem Header="Основные"><ScrollViewer><StackPanel Orientation="Vertical" Margin="10,10,10,40" VerticalAlignment="Top"><StackPanel Orientation="Horizontal" Margin="5"><Label Content="Ответственный кладовщик:" VerticalAlignment="Center" Width="200"/><ComboBox ItemsSource="{Binding Employees, UpdateSourceTrigger=PropertyChanged}" SelectedItem="{Binding SelecteStorekeeper}" Style="{StaticResource MaterialDesignOutlinedComboBox}" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" DisplayMemberPath="GetFioAndPost" Width="400" materialDesign:HintAssist.Hint="Сотрудник"/></StackPanel><StackPanel Orientation="Horizontal" Margin="5"><Label Content="Ответственный бухгалтер:" VerticalAlignment="Center" Width="200"/><ComboBox ItemsSource="{Binding Employees, UpdateSourceTrigger=PropertyChanged}" SelectedItem="{Binding SelectedAccountant}" Style="{StaticResource MaterialDesignOutlinedComboBox}" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" DisplayMemberPath="GetFioAndPost" Width="400" materialDesign:HintAssist.Hint="Сотрудник"/></StackPanel><StackPanel Orientation="Horizontal" Margin="5"><Label Content="Ставка НДС: " VerticalAlignment="Center" Width="200"/><TextBox Style="{StaticResource MaterialDesignOutlinedTextBox}" Width="100" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="НДС" materialDesign:TextFieldAssist.SuffixText="%"><Binding Path="SelectNds"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></TextBox></StackPanel><StackPanel Orientation="Horizontal" Margin="5"><Label Content="Отпуск продукции разрешает:" VerticalAlignment="Center" Width="200"/><ComboBox ItemsSource="{Binding Employees, UpdateSourceTrigger=PropertyChanged}" SelectedItem="{Binding SelectedProductReleaseApprover}" Style="{StaticResource MaterialDesignOutlinedComboBox}" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" DisplayMemberPath="GetFioAndPost" Width="400" materialDesign:HintAssist.Hint="Сотрудник"/></StackPanel><StackPanel Orientation="Horizontal" Margin="5"><Label Content="Сдаёт продукцию:" VerticalAlignment="Center" Width="200"/><ComboBox ItemsSource="{Binding Employees, UpdateSourceTrigger=PropertyChanged}" SelectedItem="{Binding SelectedProductSubmitter}" Style="{StaticResource MaterialDesignOutlinedComboBox}" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" DisplayMemberPath="GetFioAndPost" Width="400" materialDesign:HintAssist.Hint="Сотрудник"/></StackPanel><StackPanel Orientation="Horizontal"><TextBox Style="{StaticResource MaterialDesignOutlinedTextBox}" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="валюта" Width="300" Text="{Binding SelectCurrency}"></TextBox></StackPanel></StackPanel></ScrollViewer></TabItem><TabItem Header="Пользователя"><ScrollViewer><StackPanel Orientation="Vertical" Margin="5"><StackPanel Orientation="Horizontal" Margin="5"><TextBox Style="{StaticResource MaterialDesignOutlinedTextBox}" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Имя пользоватя (логин)" Width="300"><TextBox.Text><Binding Path="CurrentUser.Login" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"><Binding.ValidationRules><validators:LoginCheck/></Binding.ValidationRules></Binding></TextBox.Text></TextBox></StackPanel><GroupBox Header="Смена пароля" materialDesign:ElevationAssist.Elevation="Dp5"><StackPanel Orientation="Vertical"><StackPanel Orientation="Horizontal" Margin="5"><PasswordBox materialDesign:HintAssist.Hint="Старый пароль" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" Width="300"><local:PasswordBoxHelper.BoundPassword><Binding Path="OldPasswordString" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"></Binding></local:PasswordBoxHelper.BoundPassword></PasswordBox><TextBlock Text="{Binding Message}" Margin="10,0,0,0" Foreground="Yellow" VerticalAlignment="Center"/></StackPanel><StackPanel Orientation="Horizontal" Margin="5"><PasswordBox x:Name="newPass" materialDesign:HintAssist.Hint="Новый пароль" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" Width="300"><local:PasswordBoxHelper.BoundPassword><Binding Path="NewPasswordString" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"><Binding.ValidationRules><validators:PasswordCheck/></Binding.ValidationRules></Binding></local:PasswordBoxHelper.BoundPassword></PasswordBox><PasswordBox materialDesign:HintAssist.Hint="новый пароль еще раз" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" Width="300" Margin="10,0,0,0"><local:PasswordBoxHelper.BoundPassword><Binding Path="NewSecondPasswordString" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"><Binding.ValidationRules><validators:PasswordCheck/></Binding.ValidationRules></Binding></local:PasswordBoxHelper.BoundPassword></PasswordBox><Button Content="Изменить" Style="{StaticResource MaterialDesignPaperButton}" Margin="25,0,0,0" Command="{Binding ChangePassword}"/></StackPanel></StackPanel></GroupBox></StackPanel></ScrollViewer></TabItem><TabItem Header="Темы"><ScrollViewer><StackPanel Orientation="Horizontal" Margin="5" VerticalAlignment="Top"><Label Content="Тема:" VerticalAlignment="Center"/><RadioButton Content="Тёмная" GroupName="ThemeRB" IsChecked="{Binding IsDarkTheme}" Style="{StaticResource MaterialDesignChoiceChipPrimaryOutlineRadioButton}"/><RadioButton Content="Светлая" GroupName="ThemeRB" IsChecked="{Binding IsLightTheme}" Style="{StaticResource MaterialDesignChoiceChipSecondaryOutlineRadioButton}"/><RadioButton Content="Следовать системе" GroupName="ThemeRB" IsChecked="{Binding IsInheritTheme}" Style="{StaticResource MaterialDesignChoiceChipOutlineRadioButton}"/></StackPanel></ScrollViewer></TabItem></TabControl><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35"><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="5" Cursor="Hand" IsCancel="True" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="5" DockPanel.Dock="Right" IsDefault="True" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/></DockPanel></Grid></Page>namespace VacTrack{public partial class SettingsPage : Page,ICachedPage{private SettingViewModel ThisViewModel=>(SettingViewModel)DataContext;public SettingsPage()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.ReloadData();}public class SettingViewModel : INotifyPropertyChanged{#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion private ObservableCollection<Employee>\_employees=null!;public ObservableCollection<Employee>Employees{get=>\_employees;set=>SetProperty(ref \_employees,value);}private ObservableCollection<Users>\_users=null!;private Users \_currentUser=null!;public Users CurrentUser{get=>\_currentUser;set=>SetProperty(ref \_currentUser,value);}private string? \_message;public string? Message{get=>\_message;set=>SetProperty(ref \_message,value);}private Employee? \_selectedAccountant;public Employee? SelectedAccountant{get=>\_selectedAccountant;set=>SetProperty(ref \_selectedAccountant,value);}private Employee? \_selecteStorekeeper;public Employee? SelecteStorekeeper{get=>\_selecteStorekeeper;set=>SetProperty(ref \_selecteStorekeeper,value);}private Employee? \_selectedProductReleaseApprover;public Employee? SelectedProductReleaseApprover{get=>\_selectedProductReleaseApprover;set=>SetProperty(ref \_selectedProductReleaseApprover,value);}private Employee? \_selectedProductSubmitter;public Employee? SelectedProductSubmitter{get=>\_selectedProductSubmitter;set=>SetProperty(ref \_selectedProductSubmitter,value);}private double \_selectNds;public double SelectNds{get=>\_selectNds;set{SetProperty(ref \_selectNds,value);}}private string \_selectCurrency=Properties.Settings.Default.Currency;public string SelectCurrency{get=>\_selectCurrency;set{SetProperty(ref \_selectCurrency,value);}}public bool IsDarkTheme{get=>SelectTheme==BaseTheme.Dark;set{if(!value)return;SelectTheme=BaseTheme.Dark;}}public bool IsLightTheme{get=>SelectTheme==BaseTheme.Light;set{if(!value)return;SelectTheme=BaseTheme.Light;}}public bool IsInheritTheme{get=>SelectTheme==BaseTheme.Inherit;set{if(!value)return;SelectTheme=BaseTheme.Inherit;}}private readonly PaletteHelper \_paletteHelper=new();private readonly Theme \_theme;private readonly DatabaseContext Db=new();public string? OldPasswordString{get;set;}public string? NewPasswordString{get;set;}public string? NewSecondPasswordString{get;set;}private BaseTheme \_thisTheme;private BaseTheme SelectTheme{get=>\_thisTheme;set{SetProperty(ref \_thisTheme,value);\_theme.SetBaseTheme(\_thisTheme);\_paletteHelper.SetTheme(\_theme);OnPropertyChanged(nameof(IsDarkTheme));OnPropertyChanged(nameof(IsLightTheme));OnPropertyChanged(nameof(IsInheritTheme));}}public ICommand CancelCommand{get;set;}public ICommand SaveCommand{get;set;}public ICommand ChangePassword{get;set;}public SettingViewModel(){\_theme=\_paletteHelper.GetTheme();CancelCommand=new RelayCommand(Cancel);SaveCommand=new RelayCommand(Save);ChangePassword=new RelayCommand(ChangePass);Init();if(\_currentUser==null||\_users==null||\_employees==null){Properties.Settings.Default.LogInUserId=-1;Properties.Settings.Default.Save();#if!DEBUG MessageBox.Show("Не удалось загрузить пользователя. Возможно, сохранённый ID недействителен.\n"+"Сохранённый пользователь не найден. Выполните вход заново.");#endif}}private void Init(){LoadData();SelectTheme=Properties.Settings.Default.AppTheme switch{"Dark"=>BaseTheme.Dark,"Light"=>BaseTheme.Light,"Inherit"=>BaseTheme.Inherit,\_=>BaseTheme.Inherit,};SelectedAccountant=Employees.FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);SelecteStorekeeper=Employees.FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleStorekeeper);SelectNds=Properties.Settings.Default.Nds;SelectedProductReleaseApprover=Employees.FirstOrDefault(e=>e.Id==Properties.Settings.Default.ProductReleaseApprover);SelectedProductSubmitter=Employees.FirstOrDefault(e=>e.Id==Properties.Settings.Default.ProductSubmitter);SelectCurrency=Properties.Settings.Default.Currency;var tempUsr=\_users.FirstOrDefault(u=>u.Id==Properties.Settings.Default.LogInUserId);if(tempUsr!=null)CurrentUser=tempUsr;}private void Cancel(object obj)=>Init();private void LoadData(){var emplDbSet=Db.Set<Employee>();emplDbSet.Include(m=>m.Post).Load();Employees=emplDbSet.Local.ToObservableCollection();var usersDbSet=Db.Set<Users>();usersDbSet.Load();\_users=usersDbSet.Local.ToObservableCollection();}public void ReloadData()=>LoadData();private void Save(object obj){if(SelectedAccountant==null||SelecteStorekeeper==null||SelectedProductReleaseApprover==null||SelectedProductSubmitter==null)return;Properties.Settings.Default.ResponsibleAccountant=SelectedAccountant.Id;Properties.Settings.Default.ResponsibleStorekeeper=SelecteStorekeeper.Id;Properties.Settings.Default.AppTheme=SelectTheme switch{BaseTheme.Dark=>"Dark",BaseTheme.Light=>"Light",BaseTheme.Inherit=>"Inherit",\_=>"Inherit"};Properties.Settings.Default.Nds=SelectNds;Properties.Settings.Default.ProductReleaseApprover=SelectedProductReleaseApprover.Id;Properties.Settings.Default.ProductSubmitter=SelectedProductSubmitter.Id;Properties.Settings.Default.Currency=SelectCurrency;Properties.Settings.Default.Save();Db.SaveChanges();}private void ChangePass(object obj){if(OldPasswordString==null||NewPasswordString==null||NewSecondPasswordString==null){Message="поля для пароля пусты";return;}if(NewPasswordString!=NewSecondPasswordString){Message="пароли не совпадают";return;}string oldPassHash=CreateUserViewModel.HashPassword(OldPasswordString);if(oldPassHash==CurrentUser.Password){string newPassHash=CreateUserViewModel.HashPassword(NewPasswordString);CurrentUser.Password=newPassHash;Db.SaveChanges();Message="Пароль изменён";OldPasswordString=string.Empty;OnPropertyChanged(nameof(OldPasswordString));NewPasswordString=string.Empty;OnPropertyChanged(nameof(NewPasswordString));NewSecondPasswordString=string.Empty;OnPropertyChanged(nameof(NewSecondPasswordString));return;}else{Message="Неверный старый пароль";return;}}}}<Page x:Class="VacTrack.NotFoundPage" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="NotFoundPage"><Page.Resources><DrawingImage x:Key="status\_notfound\_borderless\_svgrepo\_comDrawingImage"><DrawingImage.Drawing><DrawingGroup ClipGeometry="M0,0 V22 H22 V0 H0 Z"><GeometryDrawing Brush="{DynamicResource MaterialDesignBody}" Geometry="F1 M22,22z M0,0z M12.822,11.29C13.638,10.709 14.243,9.942 14.505,8.968 15.108,6.725 13.532,4.415 10.975,4.415 9.825,4.415 8.89,4.825 8.2,5.504 7.78,5.917 7.528,6.339 7.4,6.671A1.179,1.179,0,0,0,9.6,7.518C9.616,7.475 9.7,7.334 9.852,7.184 10.116,6.925 10.465,6.772 10.975,6.772 11.913,6.772 12.445,7.552 12.229,8.356 12.124,8.746 11.859,9.082 11.456,9.368A3.25,3.25,0,0,1,10.511,9.838A1.179,1.179,0,0,0,9.637,10.976L9.637,13.21A1.179,1.179,0,1,0,11.995,13.21L11.995,11.78A5.9,5.9,0,0,0,12.822,11.288z"/><GeometryDrawing Brush="{DynamicResource MaterialDesignBody}"><GeometryDrawing.Geometry><EllipseGeometry RadiusX="1.275" RadiusY="1.322" Center="10.825,16.711"/></GeometryDrawing.Geometry></GeometryDrawing></DrawingGroup></DrawingImage.Drawing></DrawingImage></Page.Resources><Grid><StackPanel HorizontalAlignment="Center" VerticalAlignment="Center"><Image Source="{StaticResource status\_notfound\_borderless\_svgrepo\_comDrawingImage}" Width="200" Height="200"/><TextBlock x:Name="textBlock" Text="base text" HorizontalAlignment="Center" FontSize="20" Foreground="{DynamicResource MaterialDesignBody}"/></StackPanel></Grid></Page>namespace VacTrack{public partial class NotFoundPage : Page{public NotFoundPage(string message){InitializeComponent();textBlock.Text=message;}}}namespace VacTrack{public interface ICachedPage{void OnNavigatedFromCache();}}<Page x:Class="VacTrack.HomePage" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="HomePage"><Page.Resources><DrawingImage x:Key="VacTimeLogoDrawingImage"><DrawingImage.Drawing><DrawingGroup ClipGeometry="M0,0 V1171 H1171 V0 H0 Z"><DrawingGroup Opacity="1"><DrawingGroup.ClipGeometry><RectangleGeometry RadiusX="0" RadiusY="0" Rect="0,0,1171,1171"/></DrawingGroup.ClipGeometry><GeometryDrawing Brush="#66FFFFFF"><GeometryDrawing.Pen><Pen Brush="#FF000000" Thickness="15" StartLineCap="Flat" EndLineCap="Flat" LineJoin="Miter"/></GeometryDrawing.Pen><GeometryDrawing.Geometry><EllipseGeometry RadiusX="578" RadiusY="578" Center="585.5,585.5"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF6EBFAC" Geometry="F0 M1171,1171z M0,0z M406.648,491L342.008,359.494C346.429,351.451,354.982,346,364.808,346L994.24,346C1002.71,346,1010.24,350.055,1014.99,356.331L951.135,491 406.648,491z"/><GeometryDrawing Brush="#FF6EBFAC"><GeometryDrawing.Geometry><RectangleGeometry RadiusX="33" RadiusY="33" Rect="613,467,147,537"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF475653"><GeometryDrawing.Geometry><RectangleGeometry RadiusX="21" RadiusY="21" Rect="436,531,127,472"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF475653" Geometry="F0 M1171,1171z M0,0z M122.646,346C116.225,362.487,116.411,381.495,124.716,398.683L397.327,962.822C412.586,994.398 450.554,1007.63 482.13,992.368 513.707,977.109 526.935,939.141 511.676,907.565L240.309,346 122.646,346z"/><GeometryDrawing Brush="#FF475653" Geometry="F1 M1171,1171z M0,0z M558.501,911.722C558.501,961.98 551,1002.72 467.001,1002.72 414.5,1002.72 417.411,1001 375.5,916 340,844 416.467,820.722 467.001,820.722 517.535,820.722 558.501,861.464 558.501,911.722z"/></DrawingGroup></DrawingGroup></DrawingImage.Drawing></DrawingImage></Page.Resources><Grid><Image Source="{StaticResource VacTimeLogoDrawingImage}" MaxHeight="400" MaxWidth="400"/></Grid></Page>namespace VacTrack{public partial class HomePage : Page{public HomePage(){InitializeComponent();}}}<Window x:Class="VacTrack.CreateUser" 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:VacTrack" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" Background="{DynamicResource MaterialDesign.Brush.Background}" mc:Ignorable="d" Title="Создание пользователя" MinHeight="300" MinWidth="340" Height="420" Width="750"><Window.DataContext><local:CreateUserViewModel/></Window.DataContext><Grid><StackPanel Orientation="Vertical" VerticalAlignment="Center" HorizontalAlignment="Center" Margin="0,10,0,0"><Label Content="Реристрация" HorizontalAlignment="Center" FontSize="20"/><TextBox Style="{StaticResource MaterialDesignOutlinedTextBox}" Width="300" materialDesign:HintAssist.Hint="Логин" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,10,0,0"><TextBox.Text><Binding Path="LoginString" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"><Binding.ValidationRules><validators:LoginCheck/></Binding.ValidationRules></Binding></TextBox.Text></TextBox><PasswordBox materialDesign:HintAssist.Hint="Пароль" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,10,0,0"><local:PasswordBoxHelper.BoundPassword><Binding Path="PasswordString" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" ValidatesOnDataErrors="True" NotifyOnValidationError="True"><Binding.ValidationRules><validators:PasswordCheck/></Binding.ValidationRules></Binding></local:PasswordBoxHelper.BoundPassword></PasswordBox><PasswordBox materialDesign:HintAssist.Hint="Повтор пароля" Style="{StaticResource MaterialDesignOutlinedRevealPasswordBox}" Foreground="{DynamicResource MaterialDesignBody}" local:PasswordBoxHelper.BoundPassword="{Binding SecondPasswordString, Mode=TwoWay, UpdateSourceTrigger=PropertyChanged}" Margin="0,10,0,0"/><Label Content="{Binding Message}" HorizontalAlignment="Center" Height="20" Margin="0,10,0,0" Foreground="Red"/><Button Content="Зарегистрировать" Style="{StaticResource MaterialDesignPaperButton}" Command="{Binding CreateUser}" Cursor="Hand" Margin="0,10,0,0"/></StackPanel></Grid></Window>namespace VacTrack{public partial class CreateUser : Window{private CreateUserViewModel ThisViewModel=>(CreateUserViewModel)DataContext;public CreateUser(){InitializeComponent();ThisViewModel.ThisWin=this;}}public class CreateUserViewModel : INotifyPropertyChanged{private string \_message=string.Empty;public string Message{get=>\_message;set{SetProperty(ref \_message,value);}}public Window? ThisWin{get;set;}#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion public string? LoginString{get;set;}public string? PasswordString{get;set;}public string? SecondPasswordString{get;set;}public ICommand CreateUser{get;set;}public CreateUserViewModel(){CreateUser=new RelayCommand(AddUser);}private void AddUser(object obj){if(PasswordString!=SecondPasswordString){Message="Пароли не совпадают";return;}if(LoginString==null||PasswordString==null){Message="Пароль не введен";return;}using(var db=new DatabaseContext()){var newUser=new Users{Login=LoginString,Password=HashPassword(PasswordString),Access="FULL"};db.Set<Users>().Add(newUser);db.SaveChanges();}if(ThisWin!=null){ThisWin.Close();Message="Успешно";}else Message="Успешно, закройте это окно";}public static string HashPassword(string password){var bytes=Encoding.UTF8.GetBytes(password);var hash=SHA256.HashData(bytes);return Convert.ToBase64String(hash);}}}<Application x:Class="VacTrack.App" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:local="clr-namespace:VacTrack" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" StartupUri="MainWindow.xaml"><Application.Resources><ResourceDictionary><ResourceDictionary.MergedDictionaries><materialDesign:CustomColorTheme BaseTheme="Inherit" PrimaryColor="#FFE64A1C" SecondaryColor="#03DAC6"/><ResourceDictionary Source="pack://application:,,,/MaterialDesignThemes.Wpf;component/Themes/MaterialDesign3.Defaults.xaml"/><ResourceDictionary Source="pack://application:,,,/MaterialDesignThemes.Wpf;component/Themes/MaterialDesignTheme.PopupBox.xaml"/></ResourceDictionary.MergedDictionaries><Style x:Key="CustomWindowStyle" TargetType="{x:Type Window}" BasedOn="{StaticResource MaterialDesignWindow}"><Setter Property="WindowChrome.WindowChrome"><Setter.Value><WindowChrome CaptionHeight="30" CornerRadius="15" GlassFrameThickness="1" NonClientFrameEdges="None" ResizeBorderThickness="10" UseAeroCaptionButtons="True"/></Setter.Value></Setter><Setter Property="BorderBrush" Value="{DynamicResource MaterialDesignPaper}"/><Setter Property="Template"><Setter.Value><ControlTemplate TargetType="{x:Type Window}"><Grid><Border Background="{TemplateBinding Background}" BorderBrush="{TemplateBinding BorderBrush}" BorderThickness="3,30,3,3"><AdornerDecorator><ContentPresenter/></AdornerDecorator></Border><DockPanel Height="30" VerticalAlignment="Top" LastChildFill="False"><TextBlock Margin="10,5,0,0" VerticalAlignment="Center" DockPanel.Dock="Left" FontSize="16" Foreground="{DynamicResource MaterialDesignBody}" Text="{TemplateBinding Title}"/><Button x:Name="btnClose" Style="{StaticResource MaterialDesignFlatDarkButton}" Foreground="Red" Click="CloseClick" DockPanel.Dock="Right" WindowChrome.IsHitTestVisibleInChrome="True"><materialDesign:PackIcon Kind="WindowClose"/></Button><Button x:Name="btnRestore" Style="{StaticResource MaterialDesignFlatDarkButton}" Foreground="{DynamicResource MaterialDesignBody}" Click="MaximizeRestoreClick" DockPanel.Dock="Right" WindowChrome.IsHitTestVisibleInChrome="True"><materialDesign:PackIcon Kind="WindowMaximize"/></Button><Button x:Name="btnMinimize" Style="{StaticResource MaterialDesignFlatDarkButton}" Foreground="{DynamicResource MaterialDesignBody}" Click="MinimizeClick" DockPanel.Dock="Right" WindowChrome.IsHitTestVisibleInChrome="True"><materialDesign:PackIcon Kind="WindowMinimize"/></Button></DockPanel></Grid></ControlTemplate></Setter.Value></Setter></Style></ResourceDictionary></Application.Resources></Application>namespace VacTrack{public partial class App : Application{public App(){InitializeComponent();}private void CloseClick(object sender,RoutedEventArgs e){var window=(Window)((FrameworkElement)sender).TemplatedParent;window.Close();}private void MaximizeRestoreClick(object sender,RoutedEventArgs e){var window=(Window)((FrameworkElement)sender).TemplatedParent;if(window.WindowState==WindowState.Normal){window.WindowState=WindowState.Maximized;}else{window.WindowState=WindowState.Normal;}}private void MinimizeClick(object sender,RoutedEventArgs e){var window=(Window)((FrameworkElement)sender).TemplatedParent;window.WindowState=WindowState.Minimized;}}}<Window x:Class="VacTrack.AboutProgram" 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:VacTrack" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" mc:Ignorable="d" Title="О программе" Height="300" Width="500" ResizeMode="NoResize"><Window.Resources><DrawingImage x:Key="VacTimeLogoDrawingImage"><DrawingImage.Drawing><DrawingGroup ClipGeometry="M0,0 V1171 H1171 V0 H0 Z"><DrawingGroup Opacity="1"><DrawingGroup.ClipGeometry><RectangleGeometry RadiusX="0" RadiusY="0" Rect="0,0,1171,1171"/></DrawingGroup.ClipGeometry><GeometryDrawing Brush="#66FFFFFF"><GeometryDrawing.Pen><Pen Brush="#FF000000" Thickness="15" StartLineCap="Flat" EndLineCap="Flat" LineJoin="Miter"/></GeometryDrawing.Pen><GeometryDrawing.Geometry><EllipseGeometry RadiusX="578" RadiusY="578" Center="585.5,585.5"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF6EBFAC" Geometry="F0 M1171,1171z M0,0z M406.648,491L342.008,359.494C346.429,351.451,354.982,346,364.808,346L994.24,346C1002.71,346,1010.24,350.055,1014.99,356.331L951.135,491 406.648,491z"/><GeometryDrawing Brush="#FF6EBFAC"><GeometryDrawing.Geometry><RectangleGeometry RadiusX="33" RadiusY="33" Rect="613,467,147,537"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF475653"><GeometryDrawing.Geometry><RectangleGeometry RadiusX="21" RadiusY="21" Rect="436,531,127,472"/></GeometryDrawing.Geometry></GeometryDrawing><GeometryDrawing Brush="#FF475653" Geometry="F0 M1171,1171z M0,0z M122.646,346C116.225,362.487,116.411,381.495,124.716,398.683L397.327,962.822C412.586,994.398 450.554,1007.63 482.13,992.368 513.707,977.109 526.935,939.141 511.676,907.565L240.309,346 122.646,346z"/><GeometryDrawing Brush="#FF475653" Geometry="F1 M1171,1171z M0,0z M558.501,911.722C558.501,961.98 551,1002.72 467.001,1002.72 414.5,1002.72 417.411,1001 375.5,916 340,844 416.467,820.722 467.001,820.722 517.535,820.722 558.501,861.464 558.501,911.722z"/></DrawingGroup></DrawingGroup></DrawingImage.Drawing></DrawingImage></Window.Resources><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><Image Source="{StaticResource VacTimeLogoDrawingImage}" HorizontalAlignment="Left" Width="160" Stretch="UniformToFill"/><TextBlock Text="ПС «Учёт выпуска и реализации продукции в ООО «Вактайм» г. Сморгонь»" TextWrapping="WrapWithOverflow" Margin="165,5,0,0" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" FontSize="14" Width="299" HorizontalAlignment="Left"/><Label Content="Разработал: Грачёв Денис Сергеевич" Margin="165,45,5,0" VerticalAlignment="Top"/><ScrollViewer Foreground="{DynamicResource MaterialDesignBody}" Width="300" Margin="165,100,0,50"><TextBlock Text=" Целью программного средства «Учёт выпуска и реализации продукции в ООО „Вактайм“ г. Сморгонь» является автоматизация процессов регистрации, хранения, обработки и анализа информации, связанной с выпуском и реализацией вакуумных установок. Программное обеспечение должно обеспечить удобный интерфейс и надёжный доступ к данным, снизить количество ручной работы, повысить точность учёта, а также ускорить получение отчётной информации. " TextWrapping="WrapWithOverflow"/></ScrollViewer><Button Style="{StaticResource MaterialDesignIconForegroundButton}" Foreground="{StaticResource MaterialDesign.Brush.Secondary}" Width="30" Height="30" VerticalAlignment="Bottom" HorizontalAlignment="Right" Margin="0,0,70,10" Click="GitHubOpen"><materialDesign:PackIcon Kind="Github" Width="30" Height="30"/></Button><Button Content="OK" Style="{StaticResource MaterialDesignOutlinedSecondaryButton}" VerticalAlignment="Bottom" HorizontalAlignment="Right" Margin="0,0,10,8" Height="32" Click="OK\_BtnClick"/></Grid></Window>namespace VacTrack{public partial class AboutProgram : Window{public AboutProgram(){InitializeComponent();}private void GitHubOpen(object sender,RoutedEventArgs e){try{Process.Start(new ProcessStartInfo("https://github.com/maldavan5916/VacTimeTP"){UseShellExecute=true});}catch(Exception ex){MessageBox.Show(ex.Message);}}private void OK\_BtnClick(object sender,RoutedEventArgs e)=>Close();}}namespace VacTrack.ViewTables{public abstract class BaseViewModel<T>: INotifyPropertyChanged where T : BaseModel{#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion #region properties protected DatabaseContext Db;protected DbSet<T>DbSet;private System.Timers.Timer? \_resetTimer;private ObservableCollection<T>\_Items;public ObservableCollection<T>Items{get=>\_Items;set{if(\_Items!=value){\_Items=value;OnPropertyChanged(nameof(Items));}}}private T? \_SelectedItem;public T? SelectedItem{get=>\_SelectedItem;set{if(!EqualityComparer<T?>.Default.Equals(\_SelectedItem,value)){\_SelectedItem=value;OnPropertyChanged(nameof(SelectedItem));}}}private string? \_Message;public string? Message{get=>\_Message;set{\_Message=value;OnPropertyChanged();StartResetTimer();}}private Brush? \_MessageBrush;public Brush? MessageBrush{get=>\_MessageBrush;set{\_MessageBrush=value;OnPropertyChanged();}}private string? \_SearchText;public string? SearchText{get=>\_SearchText;set{\_SearchText=value;OnPropertyChanged();Search();}}private string? \_TableName;public string? TableName{get=>\_TableName;set{\_TableName=value;OnPropertyChanged();}}public bool? IsAllSelected{get{var selected=Items.Select(item=>item.IsSelected).Distinct().ToList();return selected.Count==1 ? selected.Single(): null;}set{if(value.HasValue){SelectAll(value.Value,Items);OnPropertyChanged();}}}private string \_currency=Properties.Settings.Default.Currency;public string Currency{get=>\_currency;set{\_currency=value;OnPropertyChanged(nameof(Currency));}}#endregion public ICommand AddCommand{get;}public ICommand DeleteCommand{get;}public ICommand SaveCommand{get;}public ICommand CancelCommand{get;}public ICommand MessageToClipboardCommand{get;}public BaseViewModel(DatabaseContext db){Db=db;Db.Database.EnsureCreated();LoadData();if(DbSet==null||\_Items==null)throw new Exception("Data loading error");AddCommand=new RelayCommand(AddItem);DeleteCommand=new RelayCommand(DeleteItem);SaveCommand=new RelayCommand(SaveChanges);CancelCommand=new RelayCommand(CancelChanges);MessageToClipboardCommand=new RelayCommand(MessageToClipboard);foreach(var model in Items){model.PropertyChanged+=(sender,args)=>{if(args.PropertyName==nameof(Employee.IsSelected))OnPropertyChanged(nameof(IsAllSelected));};}}protected abstract T CreateNewItem();protected abstract bool FilterItem(T item,string? searchText);protected virtual void LoadData(){DbSet=Db.Set<T>();DbSet.Load();Items=DbSet.Local.ToObservableCollection();}protected void AddItem(object obj){try{var newItem=CreateNewItem();Items.Add(newItem);Message="Новая запись добавлена";MessageBrush=Brushes.Green;}catch(Exception ex){Message=$"Ошибка при добавлении: {ex.Message}";MessageBrush=Brushes.Red;}}protected void DeleteItem(object obj){try{if(SelectedItem!=null){Items.Remove(SelectedItem);Message="Запись удалена";MessageBrush=Brushes.Green;}else{Message="Выберите строку для удаления.";MessageBrush=Brushes.Orange;}}catch(Exception ex){System.Windows.MessageBox.Show($"Ошибка при удалении: {ex.Message}");MessageBrush=Brushes.Red;}}protected void SaveChanges(object obj){try{var changedEntries=Db.ChangeTracker.Entries().Where(e=>e.State==EntityState.Modified||e.State==EntityState.Added||e.State==EntityState.Deleted).ToList();if(changedEntries.Count!=0){Db.SaveChanges();Message=$"Изменения для таблицы \"{TableName}\" успешно сохранены.";MessageBrush=Brushes.Green;}else{Message=$"Нет изменений для сохранения в таблице \"{TableName}\".";MessageBrush=Brushes.Orange;}}catch(DbUpdateException dbEx){Message=$"Ошибка обновления базы данных: {dbEx.Message}";MessageBrush=Brushes.Red;}catch(Exception ex){Message=$"Ошибка сохранения изменений: {ex.Message}";MessageBrush=Brushes.Red;}}protected void CancelChanges(object obj){try{var changedEntries=Db.ChangeTracker.Entries().Where(e=>e.State==EntityState.Modified||e.State==EntityState.Added||e.State==EntityState.Deleted);if(changedEntries.ToList().Count==0){Message="Нечего отменять";MessageBrush=Brushes.Orange;return;}Db.ChangeTracker.Clear();LoadData();Message="Изменения отменены.";MessageBrush=Brushes.Green;}catch(Exception ex){Message=$"Ошибка при отмене изменений: {ex.Message}";MessageBrush=Brushes.Red;}}private void Search(){try{if(string.IsNullOrWhiteSpace(SearchText)){LoadData();Message=string.Empty;return;}var filteredItems=DbSet.Local.Where(item=>FilterItem(item,SearchText)).ToList();Items=[..filteredItems];Message=$"Найдено записей: {filteredItems.Count}";MessageBrush=filteredItems.Count!=0 ? Brushes.Green : Brushes.Orange;}catch(Exception ex){Message=$"Ошибка поиска: {ex.Message}";MessageBrush=Brushes.Red;}}private void StartResetTimer(){\_resetTimer?.Stop();\_resetTimer=new System.Timers.Timer(10000);\_resetTimer.Elapsed+=ResetMessage;\_resetTimer.Start();}private void ResetMessage(object? sender,ElapsedEventArgs e){App.Current.Dispatcher.Invoke(()=>{Message=string.Empty;});\_resetTimer?.Stop();}public void OpenFromCache()=>LoadData();private static void SelectAll(bool select,IEnumerable<T>models){foreach(var model in models){model.IsSelected=select;}}private void MessageToClipboard(object obj)=>Clipboard.SetText(Message);}}<Page x:Class="VacTrack.ViewTables.ContractViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:sys="clr-namespace:System;assembly=System.Runtime" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="ContractViewTable"><Page.DataContext><local:ContractViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter" Length="720"/><converter:ContractStatusConverter x:Key="ContractStatusConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Номер"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridComboBoxColumn Header="Контрагент" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ContractCounterpartie, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Counterpartie}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><DataGridTemplateColumn Header="Дата заключения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Date, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding Date, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Срок (месацев)"><materialDesign:DataGridTextColumn.Binding><Binding Path="Term"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Cумма" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Summ"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridComboBoxColumn Header="Изделие" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ContractProduct, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Product}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridComboBoxColumn Header="Статус" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.Statuses, Converter={StaticResource ContractStatusConverter}}" SelectedValueBinding="{Binding Status, UpdateSourceTrigger=PropertyChanged, Converter={StaticResource ContractStatusConverter}}" IsEditable="False"/></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" Width="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/><Button Content="Печать" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding PrintCommand}" Width="95" ToolTip="Печать договора"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class ContractViewTable : Page,ICachedPage{private ContractViewModel ThisViewModel=>(ContractViewModel)DataContext;public ContractViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class ContractViewModel : BaseViewModel<Contract>{public ObservableCollection<Counterpartie>? ContractCounterpartie{get;set;}public ObservableCollection<Product>? ContractProduct{get;set;}public List<ContractStatus>Statuses{get;}=[ ContractStatus.created,ContractStatus.running,ContractStatus.completed,];public ICommand PrintCommand{get;}public ContractViewModel(): base(new DatabaseContext()){PrintCommand=new RelayCommand(PrintContract);}protected override void LoadData(){TableName="Договора";ContractCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);ContractProduct=new ObservableCollection<Product>([..Db.Products]);DbSet=Db.Set<Contract>();DbSet.Include(e=>e.Counterpartie).Include(e=>e.Product).ThenInclude(p=>p!=null ? p.Unit : null).Load();Items=DbSet.Local.ToObservableCollection();}protected override Contract CreateNewItem()=>new(){Name="ДП-0",Date=DateTime.Now};protected override bool FilterItem(Contract item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Counterpartie?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Product?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Summ.ToString().Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Status.ToString().Contains(searchText,StringComparison.CurrentCultureIgnoreCase);private void PrintContract(object obj){if(SelectedItem==null){Message="Запись не выбранна";MessageBrush=Brushes.Orange;return;}new ContractReport(SelectedItem).Show();}}}<Page x:Class="VacTrack.ViewTables.CounterpartieViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="1080" Title="CounterpartieViewTable"><Page.DataContext><local:CounterpartieViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/><converter:CounterpartieTypeConverter x:Key="CounterpartieTypeConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="ФИО/Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Юридический адрес"><materialDesign:DataGridTextColumn.Binding><Binding Path="LegalAddress"><Binding.ValidationRules><validators:LengthRangeValidationRule MinLength="5" MaxLength="128"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Номер телефона"><materialDesign:DataGridTextColumn.Binding><Binding Path="PhoneNomber"><Binding.ValidationRules><validators:PhoneNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Почтовый адресс"><materialDesign:DataGridTextColumn.Binding><Binding Path="PostalAddress"><Binding.ValidationRules><validators:LengthRangeValidationRule MinLength="5" MaxLength="128"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="УНП"><materialDesign:DataGridTextColumn.Binding><Binding Path="Unp"><Binding.ValidationRules><validators:LengthRangeValidationRule Length="9"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTemplateColumn Header="Расчетный счет" Width="130"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding BankAccount}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><TextBox Text="{Binding BankAccount}" AcceptsReturn="True" VerticalScrollBarVisibility="Auto"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridComboBoxColumn Header="Тип" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.Types, Converter={StaticResource CounterpartieTypeConverter}}" SelectedValueBinding="{Binding Type, UpdateSourceTrigger=PropertyChanged, Converter={StaticResource CounterpartieTypeConverter}}" IsEditable="False"/><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="ОКЮЛП"><materialDesign:DataGridTextColumn.Binding><Binding Path="Okulp"><Binding.ValidationRules><validators:LengthRangeValidationRule Length="10"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="ОКПО"><materialDesign:DataGridTextColumn.Binding><Binding Path="Okpo"><Binding.ValidationRules><validators:LengthRangeValidationRule Length="10"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="ОКЕД"><materialDesign:DataGridTextColumn.Binding><Binding Path="Oked"><Binding.ValidationRules><validators:LengthRangeValidationRule Length="5"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class CounterpartieViewTable : Page,ICachedPage{private CounterpartieViewModel ThisViewModel=>(CounterpartieViewModel)DataContext;public CounterpartieViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class CounterpartieViewModel : BaseViewModel<Counterpartie>{public List<CounterpartieType>Types{get;}=[CounterpartieType.Fiz,CounterpartieType.Ur];public CounterpartieViewModel(): base(new DatabaseContext()){TableName="Контрагенты";}protected override Counterpartie CreateNewItem()=>new(){Name="Новый контрагент",LegalAddress="Юридический адрес",PhoneNomber="Номер телефона",PostalAddress="Почтовый адрес",Unp="УНП",BankAccount="Счёт"};protected override bool FilterItem(Counterpartie item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;}}<Page x:Class="VacTrack.ViewTables.DivisionViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="DivisionViewTable"><Page.DataContext><local:DivisionViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" Margin="0,20,0,35" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class DivisionViewTable : Page{public DivisionViewTable(){InitializeComponent();}}public class DivisionViewModel : BaseViewModel<Division>{public DivisionViewModel(): base(new DatabaseContext()){TableName="Подразделения";}protected override Division CreateNewItem()=>new(){Name="Новое подразделение"};protected override bool FilterItem(Division item,string filter)=>item.Name!=null&&item.Name.Contains(filter,StringComparison.CurrentCultureIgnoreCase);}}<Page x:Class="VacTrack.ViewTables.EmployeeViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:sys="clr-namespace:System;assembly=System.Runtime" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="EmployeeViewTable"><Page.DataContext><local:EmployeeViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="ФИО"><materialDesign:DataGridTextColumn.Binding><Binding Path="Fio"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTemplateColumn Header="Подразделения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Division.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.EmployDivisions, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Division, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><DataGridTemplateColumn Header="Должность"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Post.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.EmployPosts, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Post, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><DataGridTemplateColumn Header="Дата найма"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding DateHire, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding DateHire, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><DataGridTemplateColumn Header="Дата рождения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding DateOfBirth, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding DateOfBirth, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Телефон"><materialDesign:DataGridTextColumn.Binding><Binding Path="PhoneNumber"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Адрес проживания"><materialDesign:DataGridTextColumn.Binding><Binding Path="Address"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Паспортные данные"><materialDesign:DataGridTextColumn.Binding><Binding Path="PassportData"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Оклад" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Salary"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Дополнительные выплаты"><materialDesign:DataGridTextColumn.Binding><Binding Path="Bonuses"></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Банковские реквизиты"><materialDesign:DataGridTextColumn.Binding><Binding Path="BankDetails"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTemplateColumn Header="Дата увольнения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding DateDismissal, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding DateDismissal, UpdateSourceTrigger=PropertyChanged, FallbackValue={x:Static sys:DateTime.Now}}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class EmployeeViewTable : Page,ICachedPage{private EmployeeViewModel ThisViewModel=>(EmployeeViewModel)DataContext;public EmployeeViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class EmployeeViewModel : BaseViewModel<Employee>{public ObservableCollection<Division>? EmployDivisions{get;set;}public ObservableCollection<Post>? EmployPosts{get;set;}public EmployeeViewModel(): base(new DatabaseContext()){}protected override void LoadData(){TableName="Сотрудники";EmployDivisions=new ObservableCollection<Division>([..Db.Divisions]);EmployPosts=new ObservableCollection<Post>([..Db.Posts]);DbSet=Db.Set<Employee>();DbSet.Include(e=>e.Division).Include(e=>e.Post).Load();Items=DbSet.Local.ToObservableCollection();}protected override Employee CreateNewItem()=>new(){Fio="Новый сотрудник",DateHire=DateTime.Now,DateOfBirth=DateTime.Now,PhoneNumber="+375",Address="Адрес",BankDetails="номер счёта",PassportData="паспорт"};protected override bool FilterItem(Employee item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Fio?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Division?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Post?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;}}<Page x:Class="VacTrack.ViewTables.LocationViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="LocationViewTable"><Page.DataContext><local:LocationViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridComboBoxColumn Header="Кладовщик" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.LocalEmployee, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Employee}" DisplayMemberPath="Fio"></materialDesign:DataGridComboBoxColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class LocationViewTable : Page,ICachedPage{private LocationViewModel ThisViewModel=>(LocationViewModel)DataContext;public LocationViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class LocationViewModel : BaseViewModel<Location>{public ObservableCollection<Employee>? LocalEmployee{get;set;}public LocationViewModel(): base(new DatabaseContext()){}protected override void LoadData(){TableName="Места хранения";LocalEmployee=new ObservableCollection<Employee>([..Db.Employees]);DbSet=Db.Set<Location>();DbSet.Include(e=>e.Employee).Load();Items=DbSet.Local.ToObservableCollection();}protected override Location CreateNewItem()=>new(){Name="Новый склад"};protected override bool FilterItem(Location item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Employee?.Fio?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;}}<Page x:Class="VacTrack.ViewTables.MaterialViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="850" Title="MaterialViewTable"><Page.DataContext><local:MaterialViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTemplateColumn Header="Единица измерения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Unit.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.MaterialUnit, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Unit, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><DataGridTemplateColumn Header="Место хранения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Location.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.MaterialLocation, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Location, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn Header="Цена за единицу" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Price"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn Header="Стоимость" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}" IsReadOnly="True"><materialDesign:DataGridTextColumn.Binding><Binding Path="GetSum"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class MaterialViewTable : Page,ICachedPage{private MaterialViewModel ThisViewModel=>(MaterialViewModel)DataContext;public MaterialViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class MaterialViewModel : BaseViewModel<Material>{public ObservableCollection<Unit>? MaterialUnit{get;set;}public ObservableCollection<Location>? MaterialLocation{get;set;}public MaterialViewModel(): base(new DatabaseContext()){}protected override void LoadData(){TableName="Материалы\\Комплектующие";MaterialUnit=new ObservableCollection<Unit>([..Db.Units]);MaterialLocation=new ObservableCollection<Location>([..Db.Locations]);DbSet=Db.Set<Material>();DbSet.Include(e=>e.Unit).Include(e=>e.Location).Load();Items=DbSet.Local.ToObservableCollection();}protected override Material CreateNewItem()=>new(){Name="Новый материал"};protected override bool FilterItem(Material item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Unit?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Location?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;}}<Page x:Class="VacTrack.ViewTables.PostViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="PostViewTable"><Page.DataContext><local:PostViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" Margin="0,20,0,35" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page><Page x:Class="VacTrack.ViewTables.PostViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="PostViewTable"><Page.DataContext><local:PostViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" Margin="0,20,0,35" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class PostViewTable : Page{public PostViewTable(){InitializeComponent();}}public class PostViewModel : BaseViewModel<Post>{public PostViewModel(): base(new DatabaseContext()){TableName="Должности";}protected override Post CreateNewItem()=>new(){Name="Новая должность"};protected override bool FilterItem(Post item,string filter)=>item.Name!=null&&item.Name.Contains(filter,StringComparison.CurrentCultureIgnoreCase);}}<Page x:Class="VacTrack.ViewTables.ProductViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="850" Title="ProductViewTable"><Page.DataContext><local:ProductViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Grid.ColumnDefinitions><ColumnDefinition Width="1\*"/><ColumnDefinition Width="400"/></Grid.ColumnDefinitions><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center" Grid.ColumnSpan="2"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}" Grid.ColumnSpan="2"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Серийный номер"><materialDesign:DataGridTextColumn.Binding><Binding Path="SerialNo"><Binding.ValidationRules><validators:LengthRangeValidationRule MinLength="5" MaxLength="10"/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTemplateColumn Header="ед.изм."><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Unit.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.ProductUnit, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Unit, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><DataGridTemplateColumn Header="Место хранения"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Location.Name, UpdateSourceTrigger=PropertyChanged}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><ComboBox ItemsSource="{Binding DataContext.ProductLocation, RelativeSource={RelativeSource AncestorType=Page}}" DisplayMemberPath="Name" SelectedValue="{Binding Location, UpdateSourceTrigger=PropertyChanged}" IsEditable="True"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Цена" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Price"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Cумма НДС" IsReadOnly="True" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="SummNds"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Цена с НДС" IsReadOnly="True" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Summ"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><Label Content="Материалы и Комлектуюшие для" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center" Grid.Column="1"/><Label Content="{Binding SelectedItem.Name}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center" Grid.Column="1" Margin="0,15,0,0"/><DataGrid Grid.Column="1" Margin="5,40,0,35" SelectedItem="{Binding SelectedMaterial}" ItemsSource="{Binding ProdMater}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><materialDesign:DataGridComboBoxColumn Header="Наименование" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ProductMaterials, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Material}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><materialDesign:DataGridTextColumn Header="ед.изм." Binding="{Binding Material.Unit.Name}" IsReadOnly="True"></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Quantity"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><materialDesign:PopupBox ToolTip="Добавить или удалить материал" Grid.Column="1" VerticalAlignment="Top" HorizontalAlignment="Right" Margin="0,46,7,0" Width="30" Height="30" PlacementMode="BottomAndAlignCentres" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Background="Transparent"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="Pencil" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel><Button Style="{StaticResource MaterialDesignIconButton}" ToolTip="Добавить материал к изделию" Command="{Binding AddMaterialCommand}" Margin="0,10,0,0" Width="30" Height="30" Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Foreground="{DynamicResource MaterialDesignBody}"><materialDesign:PackIcon Kind="Plus" Width="25" Height="25"/></Button><Button Style="{StaticResource MaterialDesignIconButton}" ToolTip="Удалить выбраный материал" Command="{Binding DeleteMaterialCommand}" Margin="0,10,0,0" Width="30" Height="30" Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Foreground="{DynamicResource MaterialDesignBody}"><materialDesign:PackIcon Kind="DeleteForeverOutline" Width="25" Height="25"/></Button></StackPanel></materialDesign:PopupBox><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel" Grid.ColumnSpan="2"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class ProductViewTable : Page,ICachedPage{private ProductViewModel ThisViewModel=>(ProductViewModel)DataContext;public ProductViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class ProductViewModel : BaseViewModel<Product>{public ObservableCollection<Unit>? ProductUnit{get;set;}public ObservableCollection<Location>? ProductLocation{get;set;}public ObservableCollection<Material>? ProductMaterials{get;set;}public ObservableCollection<Product\_Material>? ProdMater{get;set;}private Product\_Material? \_SelectedMaterial;public Product\_Material? SelectedMaterial{get=>\_SelectedMaterial;set{if(!EqualityComparer<Product\_Material?>.Default.Equals(\_SelectedMaterial,value)){\_SelectedMaterial=value;OnPropertyChanged(nameof(SelectedItem));}}}private Product? LastSelectedMaterial;public ICommand AddMaterialCommand{get;}public ICommand DeleteMaterialCommand{get;}public ProductViewModel(): base(new DatabaseContext()){AddMaterialCommand=new RelayCommand(AddMaterial);DeleteMaterialCommand=new RelayCommand(DeleteMaterial);PropertyChanged+=OnPropertyChangedHandler;}private void OnPropertyChangedHandler(object? sender,PropertyChangedEventArgs e){if(e.PropertyName==nameof(SelectedItem)&&sender is ProductViewModel pvm)LoadProdMater(pvm.SelectedItem);}private void LoadProdMater(Product? product){if(product==null)ProdMater?.Clear();else{if(LastSelectedMaterial==product)return;var productMaterials=Db.Product\_Materials.Include(e=>e.Material).Where(e=>e.ProductId==product.Id).ToList();ProdMater=[..productMaterials];LastSelectedMaterial=product;OnPropertyChanged(nameof(ProdMater));}}private void AddMaterial(object obj){try{if(ProdMater!=null||ProdMater?.Count==0){if(SelectedItem==null)throw new Exception();var newPM=new Product\_Material(){ProductId=SelectedItem.Id,Product=SelectedItem};ProdMater.Add(newPM);Db.Product\_Materials.Add(newPM);Message="Материал добавлен";MessageBrush=Brushes.Green;}else{Message="Выберите изделие";MessageBrush=Brushes.Orange;}}catch(Exception ex){Message=$"Ошибка при добавлении материала: {ex.Message}";MessageBrush=Brushes.Red;}}private void DeleteMaterial(object obj){try{if(SelectedMaterial!=null&&ProdMater!=null){Db.Product\_Materials.Remove(SelectedMaterial);ProdMater.Remove(SelectedMaterial);Message="Материал удален";MessageBrush=Brushes.Green;}else{Message="Выберите материал для удаления.";MessageBrush=Brushes.Orange;}}catch(Exception ex){System.Windows.MessageBox.Show($"Ошибка при удалении материала: {ex.Message}");MessageBrush=Brushes.Red;}}protected override void LoadData(){TableName="Продукция";ProductUnit=new ObservableCollection<Unit>([..Db.Units]);ProductLocation=new ObservableCollection<Location>([..Db.Locations]);ProductMaterials=new ObservableCollection<Material>([..Db.Materials]);DbSet=Db.Set<Product>();DbSet.Include(e=>e.Unit).Include(e=>e.Location).Load();Items=DbSet.Local.ToObservableCollection();foreach(var item in Items){item.Nds=Properties.Settings.Default.Nds;}}protected override Product CreateNewItem()=>new(){Name="Новое изделие",SerialNo="Серийный номер",Nds=Properties.Settings.Default.Nds};protected override bool FilterItem(Product item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Unit?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Location?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;}}<Page x:Class="VacTrack.ViewTables.ReceiptViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:sys="clr-namespace:System;assembly=System.Runtime" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="ReceiptViewTable"><Page.DataContext><local:ReceiptViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter" Length="715"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridComboBoxColumn Header="Контрагент" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ReceiptCounterpartie, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Counterpartie}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><DataGridTemplateColumn Header="Дата поступления"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Date, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding Date, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridComboBoxColumn Header="Материал" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ReceiptMaterial, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Material}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><DataGridTextColumn IsReadOnly="True" Header="Ед.Изм." Binding="{Binding Material.Unit.Name, UpdateSourceTrigger=PropertyChanged}"/><DataGridTextColumn IsReadOnly="True" Header="Цена" Binding="{Binding Material.Price, UpdateSourceTrigger=PropertyChanged}" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"/><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Cумма" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Summ" UpdateSourceTrigger="PropertyChanged"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddItemsCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/><Button Content="Печать" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding PrintCommand}" Width="95" ToolTip="Печать выбраного документа"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class ReceiptViewTable : Page,ICachedPage{private ReceiptViewModel ThisViewModel=>(ReceiptViewModel)DataContext;public ReceiptViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class ReceiptViewModel : BaseViewModel<Receipt>{public ObservableCollection<Counterpartie>? ReceiptCounterpartie{get;set;}public ObservableCollection<Material>? ReceiptMaterial{get;set;}public ICommand PrintCommand{get;}public ICommand AddItemsCommand{get;}public new ICommand DeleteCommand{get;}public ReceiptViewModel(): base(new DatabaseContext()){PrintCommand=new RelayCommand(PrintDoc);AddItemsCommand=new RelayCommand(AddItems);DeleteCommand=new RelayCommand(DeleteItem);}protected override void LoadData(){TableName="Поступления";ReceiptCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);var MaterDbSet=Db.Set<Material>();MaterDbSet.Include(m=>m.Unit).Load();ReceiptMaterial=MaterDbSet.Local.ToObservableCollection();DbSet=Db.Set<Receipt>();DbSet.Include(e=>e.Counterpartie).Include(e=>e.Material).ThenInclude(e=>e!=null ? e.Unit : null).Include(e=>e.Material!=null ? e.Material.Location : null).ThenInclude(e=>e!=null ? e.Employee : null).Load();Items=DbSet.Local.ToObservableCollection();}protected override Receipt CreateNewItem()=>new(){Date=DateTime.Now};protected override bool FilterItem(Receipt item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Counterpartie?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Material?.Name?.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Summ.ToString().Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;private void PrintDoc(object obj){var printWindow=new ReceiptsPrint();bool? dialogResult=printWindow.ShowDialog();if(dialogResult==true){if(printWindow.Items!=null&&printWindow.ReceiptsPrintVM!=null)new ViewReport.ReceiptReport(printWindow.Items,printWindow.ReceiptsPrintVM).Show();else throw new Exception("Cannot display the report because either `Items` or `ReceiptsPrintVM` is null.");}}private void AddItems(object obj){var ReceiptsWindow=new ReceiptsAdd();bool? dialogResult=ReceiptsWindow.ShowDialog();if(dialogResult==true){var newItems=ReceiptsWindow.NewItems;if(newItems==null){Message="Нет элементов для добавления";MessageBrush=Brushes.Orange;return;}foreach(var item in newItems){if(item.Material==null||item.Counterpartie==null){Message="Сведения о материале или контрагенте не были получены";MessageBrush=Brushes.Red;return;}Items.Add(new Receipt{Id=item.Id,MaterialId=item.Material.Id,Date=item.Date,Count=item.Count,CounterpartyId=item.Counterpartie.Id,Summ=item.Summ});Db.Materials.First(m=>m.Id==item.Material.Id).Count+=item.Count;}Message="Успешно добалено";MessageBrush=Brushes.Green;}}private new void DeleteItem(object obj){Receipt? tempSelectedItem=SelectedItem;base.DeleteItem(obj);if(tempSelectedItem==null){return;}Db.Materials.First(m=>m.Id==tempSelectedItem.MaterialId).Count-=tempSelectedItem.Count;}}}<Page x:Class="VacTrack.ViewTables.SaleViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:sys="clr-namespace:System;assembly=System.Runtime" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="SaleViewTable"><Page.DataContext><local:SaleViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter" Length="720"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid Margin="0,20,0,35" ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><DataGridTemplateColumn Header="Договор"><DataGridTemplateColumn.CellTemplate><!--Отображение после выбора(только Name)--><DataTemplate><TextBlock Text="{Binding Contract.Name}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><!--Выпадающий список--><DataTemplate><ComboBox ItemsSource="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.SaleContract}" SelectedItem="{Binding Contract, Mode=TwoWay, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="DisplayFull" IsEditable="False"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn Header="Изделие по договору" IsReadOnly="True" Binding="{Binding Contract.Product.Name, UpdateSourceTrigger=PropertyChanged}"/><DataGridTemplateColumn Header="Дата"><DataGridTemplateColumn.CellTemplate><DataTemplate><TextBlock Text="{Binding Date, StringFormat={}{0:dd.MM.yyyy}}"/></DataTemplate></DataGridTemplateColumn.CellTemplate><DataGridTemplateColumn.CellEditingTemplate><DataTemplate><DatePicker Width="90" SelectedDate="{Binding Date, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" materialDesign:HintAssist.Hint="Выбор даты"/></DataTemplate></DataGridTemplateColumn.CellEditingTemplate></DataGridTemplateColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Cумма" materialDesign:TextFieldAssist.SuffixText="{Binding Currency}"><materialDesign:DataGridTextColumn.Binding><Binding Path="Summ"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" Width="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/><Button Content="Печать" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding PrintCommand}" Width="95" ToolTip="Печать выбранной записи"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class SaleViewTable : Page,ICachedPage{private SaleViewModel ThisViewModel=>(SaleViewModel)DataContext;public SaleViewTable()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class SaleViewModel : BaseViewModel<Sale>{public ObservableCollection<Contract>? SaleContract{get;set;}public ICommand PrintCommand{get;}public SaleViewModel(): base(new DatabaseContext()){PrintCommand=new RelayCommand(PrintDoc);}protected override void LoadData(){TableName="Реализация";SaleContract=new ObservableCollection<Contract>([..Db.Contracts]);DbSet=Db.Set<Sale>();DbSet.Include(e=>e.Contract).ThenInclude(c=>c!=null ? c.Product : null).ThenInclude(p=>p!=null ? p.Unit : null).Include(e=>e.Contract).ThenInclude(c=>c!=null ? c.Counterpartie : null).Load();Items=DbSet.Local.ToObservableCollection();}protected override Sale CreateNewItem()=>new(){Date=DateTime.Now};protected override bool FilterItem(Sale item,string? searchText)=>string.IsNullOrWhiteSpace(searchText)||item.Contract?.Product?.Name.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Contract?.Name.Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true||item.Summ.ToString().Contains(searchText,StringComparison.CurrentCultureIgnoreCase)==true;private void PrintDoc(object obj){if(SelectedItem==null){Message="Запись не выбранна";MessageBrush=Brushes.Orange;return;}new SaleReport(SelectedItem).Show();}}}<Page x:Class="VacTrack.ViewTables.UnitViewTable" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewTables" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="UnitViewTable"><Page.DataContext><local:UnitViewModel/></Page.DataContext><Page.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter"/></Page.Resources><Grid><Label Content="{Binding TableName}" Foreground="{DynamicResource MaterialDesignBody}" VerticalAlignment="Top" HorizontalAlignment="Center"/><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><DataGrid ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" Margin="0,20,0,35" BorderThickness="3" BorderBrush="#19737171"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Наименование"><materialDesign:DataGridTextColumn.Binding><Binding Path="Name"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Описание"><materialDesign:DataGridTextColumn.Binding><Binding Path="Description"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><Button Content="Сохранить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding SaveCommand}" Width="95" ToolTip="сохраняет внесеные изменения"/><TextBox materialDesign:TextFieldAssist.HasClearButton="True" Style="{StaticResource MaterialDesignFloatingHintTextBox}" Width="200" materialDesign:HintAssist.Hint="Поиск" Foreground="{DynamicResource MaterialDesignBody}" Text="{Binding SearchText, UpdateSourceTrigger=PropertyChanged}" Margin="5,0,3,0"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Command="{Binding CancelCommand}" Width="95" ToolTip="отменяет не сохраненые изменения"/></DockPanel></Grid></Page>namespace VacTrack.ViewTables{public partial class UnitViewTable : Page{public UnitViewTable(){InitializeComponent();}}public class UnitViewModel : BaseViewModel<Unit>{public UnitViewModel(): base(new DatabaseContext()){TableName="Единицы измерения";}protected override Unit CreateNewItem()=>new(){Name="Новая единица"};protected override bool FilterItem(Unit item,string filter)=>item.Name!=null&&item.Name.Contains(filter,StringComparison.CurrentCultureIgnoreCase);}}namespace VacTrack.ViewReport{public abstract class BaseReportViewModel<T>: INotifyPropertyChanged where T : BaseModel{#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion #region properties protected DatabaseContext Db;protected DbSet<T>DbSet;private System.Timers.Timer? \_resetTimer;private bool \_isRefreshing=false;private ObservableCollection<T>\_Items;public ObservableCollection<T>Items{get=>\_Items;set{if(\_Items!=value){\_Items=value;Refresh(null);SetProperty(ref \_Items,value);}}}private FlowDocument? \_Document;public FlowDocument? Document{get=>\_Document;set=>SetProperty(ref \_Document,value);}private string? \_Message;public string? Message{get=>\_Message;set{\_Message=value;OnPropertyChanged();StartResetTimer();}}private Brush? \_MessageBrush;public Brush? MessageBrush{get=>\_MessageBrush;set{\_MessageBrush=value;OnPropertyChanged();}}#endregion public ICommand RefreshCommand{get;}public ICommand PrintCommand{get;}public ICommand RotateDocCommand{get;}public BaseReportViewModel(){Db=new DatabaseContext();Db.Database.EnsureCreated();LoadData();if(DbSet==null||\_Items==null)throw new Exception("Data loading error");RefreshCommand=new RelayCommand(Refresh);PrintCommand=new RelayCommand(PrintReport);RotateDocCommand=new RelayCommand(SwapPageDimensions);}public abstract FlowDocument CreateReport();protected void Refresh(object? obj){if(\_isRefreshing)return;try{\_isRefreshing=true;Db=new DatabaseContext();LoadData();Document=CreateReport();Document.PageWidth=793.7;Document.PageHeight=1122.52;Document.PagePadding=new Thickness(50);Document.ColumnWidth=double.PositiveInfinity;}finally{\_isRefreshing=false;}}protected virtual void LoadData(){DbSet=Db.Set<T>();DbSet.Load();Items=DbSet.Local.ToObservableCollection();}private void PrintReport(object obj){if(Document==null){Message="Документ пуст";MessageBrush=Brushes.Orange;return;}PrintDialog printDialog=new();if(printDialog.ShowDialog()==true){var foreground=Document.Foreground;var background=Document.Background;Document.PageWidth=printDialog.PrintableAreaWidth;Document.PageHeight=printDialog.PrintableAreaHeight;Document.Foreground=Brushes.Black;Document.Background=Brushes.White;printDialog.PrintDocument(((IDocumentPaginatorSource)Document).DocumentPaginator,"Печать отчёта");Document.Foreground=foreground;Document.Background=background;}}public void OpenFromCache()=>LoadData();private void StartResetTimer(){\_resetTimer?.Stop();\_resetTimer=new System.Timers.Timer(10000);\_resetTimer.Elapsed+=ResetMessage;\_resetTimer.Start();}private void ResetMessage(object? sender,ElapsedEventArgs e){App.Current.Dispatcher.Invoke(()=>{Message=string.Empty;});\_resetTimer?.Stop();}private void SwapPageDimensions(object obj){if(Document==null){Message="Документ пуст";MessageBrush=Brushes.Orange;return;}double tempWidth=Document.PageWidth;double tempHeight=Document.PageHeight;Document.PageWidth=tempHeight;Document.PageHeight=tempWidth;}public void CreateGroupedRows<TKey>(ref TableRowGroup dataGroup,Func<T,TKey>keySelector,Func<TKey,IEnumerable<string>>groupHeaderSelector,Func<T,IEnumerable<string>>rowSelector){var groupedItems=Items.GroupBy(keySelector);foreach(var group in groupedItems){dataGroup.Rows.Add(CreateRow(groupHeaderSelector(group.Key)));foreach(var item in group){dataGroup.Rows.Add(CreateRow(rowSelector(item)));}}}public void CreateGroupedRows<TKey>(ref TableRowGroup dataGroup,Func<T,TKey>keySelector,Func<T,double>getSum,Func<TKey,IEnumerable<string>>groupHeaderSelector,Func<double,IEnumerable<string>>groupTotalSelector,Func<T,IEnumerable<string>>rowSelector,ref double totalSum,bool IsGroupTotalEnabled){var groupedItems=Items.GroupBy(keySelector);foreach(var group in groupedItems){dataGroup.Rows.Add(CreateRow(groupHeaderSelector(group.Key)));double summ=0;foreach(var item in group){dataGroup.Rows.Add(CreateRow(rowSelector(item)));summ+=getSum(item);}if(IsGroupTotalEnabled)dataGroup.Rows.Add(CreateRow(groupTotalSelector(summ)));totalSum+=summ;}}public void CreateGroupedRows<TKey>(ref TableRowGroup dataGroup,Func<T,TKey>keySelector,Func<T,double>getSum,Func<T,double>getCount,Func<TKey,IEnumerable<string>>groupHeaderSelector,Func<(double,double),IEnumerable<string>>groupTotalSelector,Func<T,IEnumerable<string>>rowSelector,ref double totalSum,bool IsGroupTotalEnabled){var groupedItems=Items.GroupBy(keySelector);foreach(var group in groupedItems){dataGroup.Rows.Add(CreateRow(groupHeaderSelector(group.Key)));double summ=0;double count=0;foreach(var item in group){dataGroup.Rows.Add(CreateRow(rowSelector(item)));summ+=getSum(item);count+=getCount(item);}if(IsGroupTotalEnabled)dataGroup.Rows.Add(CreateRow(groupTotalSelector((count,summ))));totalSum+=summ;}}static public TableRow CreateRow(IEnumerable<string>columns){TableRow row=new();foreach(var column in columns)row.Cells.Add(new TableCell(new Paragraph(new Run(column ?? string.Empty))));return row;}}}<Page x:Class="VacTrack.ViewReport.ContractorContractsReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:converter="clr-namespace:VacTrack.Converter" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="ContractorContractsReport"><Page.DataContext><local:ContractorContractsReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Page.Resources><converter:InvertBooleanConverter x:Key="InvertBooleanConverter"/></Page.Resources><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><materialDesign:PopupBox StaysOpen="True" Margin="20,0,0,0" ToolTip="Выбор типа групировки" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FormatListGroupPlus" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Margin="0,5,0,0"><RadioButton Content="Групировка по контрагенту" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByCounterpartie}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Групировка по изделию" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByProduct}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Без групировки" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeNoGrouped}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><Border Margin="0,15,0,0" Background="{DynamicResource MaterialDesign.Brush.Background}" CornerRadius="15"><StackPanel><Label Content="Настройка итогов"/><ListBox Style="{StaticResource MaterialDesignFilterChipPrimaryListBox}" Height="46"><ListBoxItem Content="Итого группы" IsSelected="{Binding IsGroupTotalEnabled}" IsEnabled="{Binding IsGroupedTypeNoGrouped, Converter={StaticResource InvertBooleanConverter}, UpdateSourceTrigger=PropertyChanged}"/><ListBoxItem Content="Общие итоги" IsSelected="{Binding AreOverallTotalsEnabled}"/></ListBox></StackPanel></Border></StackPanel></materialDesign:PopupBox><materialDesign:PopupBox StaysOpen="True" Margin="10,0,0,0" ToolTip="Настиройки фильтрации" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FilterSettingsOutline" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Width="200" Background="{DynamicResource MaterialDesign.Brush.Background}"><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding ContractCounterpartie}" SelectedItem="{Binding FilterByCounterpartie, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по котрагенту" materialDesign:TextFieldAssist.HasClearButton="True"/><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding ContractProduct}" SelectedItem="{Binding FilterByProduct, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по изделию" materialDesign:TextFieldAssist.HasClearButton="True"/><Label Content="За период" Margin="0,15,0,0"/><DatePicker SelectedDate="{Binding FilterStartDate, UpdateSourceTrigger=PropertyChanged}" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="С" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><DatePicker SelectedDate="{Binding FilterEndDate, UpdateSourceTrigger=PropertyChanged}" Margin="0,5,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="По" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><Button Content="Очистить фильтр" Margin="0,15,0,0" Style="{StaticResource MaterialDesignOutlinedButton}" Command="{Binding ClearFilterCommand}"/></StackPanel></materialDesign:PopupBox><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class ContractorContractsReport : Page{private ContractorContractsReportViewModel ThisViewModel=>(ContractorContractsReportViewModel)DataContext;public ContractorContractsReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public enum ContractGroupedType{NoGrouped,GroupedByCounterpartie,GroupedByProduct}public class ContractorContractsReportViewModel : BaseReportViewModel<Contract>{public ObservableCollection<Counterpartie>? ContractCounterpartie{get;set;}public ObservableCollection<Product>? ContractProduct{get;set;}#region properties private ContractGroupedType GroupedType=ContractGroupedType.NoGrouped;public bool IsGroupedTypeNoGrouped{get=>GroupedType==ContractGroupedType.NoGrouped;set{if(value){GroupedType=ContractGroupedType.NoGrouped;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}public bool IsGroupedTypeGroupedByCounterpartie{get=>GroupedType==ContractGroupedType.GroupedByCounterpartie;set{if(value){GroupedType=ContractGroupedType.GroupedByCounterpartie;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByCounterpartie));OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}public bool IsGroupedTypeGroupedByProduct{get=>GroupedType==ContractGroupedType.GroupedByProduct;set{if(value){GroupedType=ContractGroupedType.GroupedByProduct;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByProduct));OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}private bool \_IsGroupTotalEnabled=true;public bool IsGroupTotalEnabled{get=>\_IsGroupTotalEnabled;set{SetProperty(ref \_IsGroupTotalEnabled,value);Refresh(null);}}private bool \_AreOverallTotalsEnabled=true;public bool AreOverallTotalsEnabled{get=>\_AreOverallTotalsEnabled;set{SetProperty(ref \_AreOverallTotalsEnabled,value);Refresh(null);}}private Counterpartie? \_FilterByCounterpartie;public Counterpartie? FilterByCounterpartie{get=>\_FilterByCounterpartie;set{SetProperty(ref \_FilterByCounterpartie,value);LoadData();}}private Product? \_FilterByProduct;public Product? FilterByProduct{get=>\_FilterByProduct;set{SetProperty(ref \_FilterByProduct,value);LoadData();}}private DateTime? \_FilterStartDate;public DateTime? FilterStartDate{get=>\_FilterStartDate;set{SetProperty(ref \_FilterStartDate,value);LoadData();}}private DateTime? \_FilterEndDate;public DateTime? FilterEndDate{get=>\_FilterEndDate;set{SetProperty(ref \_FilterEndDate,value);LoadData();}}#endregion public System.Windows.Input.ICommand ClearFilterCommand{get;}public ContractorContractsReportViewModel(){ClearFilterCommand=new RelayCommand(ClearFilter);}protected override void LoadData(){ContractCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);ContractProduct=new ObservableCollection<Product>([..Db.Products]);DbSet=Db.Set<Contract>();DbSet.Include(e=>e.Counterpartie).Include(e=>e.Product).ThenInclude(e=>e!=null ? e.Unit : null).Load();Items=[..DbSet.Local.Where(item=>(FilterByCounterpartie==null||item.Counterpartie?.Id==FilterByCounterpartie.Id)&&(FilterByProduct==null||item.Product?.Id==FilterByProduct.Id)&&((FilterStartDate==null&&FilterEndDate==null)||(FilterStartDate!=null&&FilterEndDate!=null&&item.Date>=FilterStartDate&&item.Date<=FilterEndDate)||(FilterStartDate!=null&&FilterEndDate==null&&item.Date==FilterStartDate)||(FilterStartDate==null&&FilterEndDate!=null&&item.Date==FilterEndDate))).ToList()];}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSum=0;switch(GroupedType){case ContractGroupedType.GroupedByCounterpartie: CreateGroupedByCounterpartie(ref dataGroup,ref totalSum);break;case ContractGroupedType.GroupedByProduct: CreateGroupedByProduct(ref dataGroup,ref totalSum);break;case ContractGroupedType.NoGrouped: CreateNoGroupedRows(ref dataGroup,ref totalSum);break;}if(AreOverallTotalsEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","","",$"{totalSum:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);doc.Blocks.Add(new Paragraph(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private void CreateGroupedByProduct(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Product?.Name,item=>item.Summ,item=>item.Count,key=>["","","",$"{key}","",""],cntTotal=>["Итого","","","",$"{cntTotal.Item1}",$"{cntTotal.Item2:N2}"],item=>[ $"{item.Name}",$"{item.Counterpartie?.Name}",$"{item.Date:dd.MM.yyyy}",string.Empty,$"{item.Count} {item.Product?.Unit?.Name}",$"{item.Summ:F2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateGroupedByCounterpartie(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Counterpartie?.Name,item=>item.Summ,key=>["",$"{key}","","","",""],total=>["Итого","","","","",$"{total:N2}"],item=>[ $"{item.Name}",String.Empty,$"{item.Date:dd.MM.yyyy}",$"{item.Product?.Name}",$"{item.Count} {item.Product?.Unit?.Name}",$"{item.Summ:N2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateNoGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){foreach(var item in Items){dataGroup.Rows.Add(CreateRow([ $"{item.Name}",$"{item.Counterpartie?.Name}",$"{item.Date:dd.MM.yyyy}",$"{item.Product?.Name}",$"{item.Count} {item.Product?.Unit?.Name}",$"{item.Summ:N2}" ]));totalSum+=item.Summ;}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Номер договора"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Контрагент"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Дата заключения"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Изделие"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Отчет по договорам с контрагентами")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filterByPost=FilterByCounterpartie==null ? string.Empty : $"\nПо контрагенту: {FilterByCounterpartie.Name}";string filterByDivision=FilterByProduct==null ? string.Empty : $"\nПо изделию: {FilterByProduct.Name}";string periodFilter=(FilterStartDate,FilterEndDate)switch{(not null,not null)=>$"\nЗа период с {FilterStartDate:dd.MM.yyyy} по {FilterEndDate:dd.MM.yyyy}",(not null,null)=>$"\nНа день {FilterStartDate:dd.MM.yyyy}",(null,not null)=>$"\nНа день {FilterEndDate:dd.MM.yyyy}",\_=>string.Empty};string filteredText=filterByPost+filterByDivision+periodFilter;string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}private void ClearFilter(object obj){FilterByCounterpartie=null;FilterByProduct=null;FilterStartDate=null;FilterEndDate=null;}}}<Window x:Class="VacTrack.ViewReport.ContractReport" 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:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" mc:Ignorable="d" Title="Договр" Height="450" Width="800" Background="{DynamicResource MaterialDesign.Brush.Background}"><Window.DataContext><local:ContarcReportViewModel/></Window.DataContext><Window.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Window.InputBindings><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Window>namespace VacTrack.ViewReport{public partial class ContractReport : Window{private ContarcReportViewModel ThisViewModel=>(ContarcReportViewModel)DataContext;public ContractReport(Contract contract){InitializeComponent();ThisViewModel.SelectContract=contract;}}public class ContarcReportViewModel : BaseReportViewModel<Contract>{private Contract? \_selectContract;public Contract? SelectContract{get=>\_selectContract;set{SetProperty(ref \_selectContract,value);Refresh(null);}}public override FlowDocument CreateReport(){FlowDocument doc=Samples.SampleContract.GenerateContract($"{SelectContract?.Name}",$"{SelectContract?.Date.ToString("'«'d'»' MMMM yyyy 'г.'", new CultureInfo("ru-RU"))}",$"{SelectContract?.Term} месяцев",$"{SelectContract?.Summ}",$"{SelectContract?.Counterpartie?.PostalAddress}",$"{SelectContract?.Counterpartie?.Name}\n"+$"{SelectContract?.Counterpartie?.LegalAddress}\n"+$"БИК: {SelectContract?.Counterpartie?.BankAccount}\n"+$"УНП: {SelectContract?.Counterpartie?.Unp}\n"+$"Тел: {SelectContract?.Counterpartie?.PhoneNomber}\n",$"ООО «Вактайм»\n"+$"231000 г. Сморгонь, Гродненская обл., ул. Пушкина 91\n"+$"p/c BY20AKBB30120877400144200000\n"+$"БИК АКВВВУ2Х в ЦБУ 423 ОЛО АСБ «Беларусбанк» г. Сморгонь ул. Советская 36\n"+$"УНН: 590976294\n"+$"ΟΚΠΟ: 300438284000\n"+$"Тел. GSM: +375 29-6299854;\n"+$"Факс: +375(1592)-4-55-23");Section newSection=new(){BreakPageBefore=true};AddAttachment(newSection);doc.Blocks.Add(newSection);return doc;}private void AddAttachment(Section section){section.FontFamily=new FontFamily("Times New Roman");section.FontSize=14;Paragraph header=new(){TextAlignment=TextAlignment.Right,FontWeight=FontWeights.Bold};header.Inlines.Add(new Run("Приложение 1"));header.Inlines.Add(new LineBreak());header.Inlines.Add(new Run($"к договору № {SelectContract?.Name} от {SelectContract?.Date.ToString("'«'d'»' MMMM yyyy 'г.'", new CultureInfo("ru-RU"))} г."));section.Blocks.Add(header);section.Blocks.Add(new Paragraph(new Run("Счет-протокол согласования договорной цены")){TextAlignment=TextAlignment.Center,FontWeight=FontWeights.Bold,Margin=new Thickness(0,20,0,10)});Table table=new(){CellSpacing=0,BorderBrush=Brushes.Black,BorderThickness=new Thickness(1)};double[] columnWidths={30,200,60,100,100,80,100,100};foreach(double width in columnWidths){table.Columns.Add(new TableColumn{});}TableRowGroup trg=new TableRowGroup();table.RowGroups.Add(trg);trg.Rows.Add(new TableRow());string[] headers=[ "Наименование","Кол-во","Цена за ед.","Сумма,\n"+Properties.Settings.Default.Currency,"ставка НДС","Сумма НДС,\n"+Properties.Settings.Default.Currency,"Всего с НДС,\n"+Properties.Settings.Default.Currency ];foreach(string h in headers){trg.Rows[0].Cells.Add(new TableCell(new Paragraph(new Run(h))));}var nds=Properties.Settings.Default.Nds;trg.Rows.Add(new TableRow());string[] row1=[ $"{SelectContract?.Product?.Name}",$"{SelectContract?.Count} {SelectContract?.Product?.Unit?.Name}",$"{SelectContract?.Product?.Price:N2}",$"{SelectContract?.Summ:N2}",$"{nds:F2}%",$"{SelectContract?.Summ \* (nds / 100):N2}",$"{SelectContract?.Summ \* (1 + nds / 100):N2}" ];foreach(string cell in row1){trg.Rows[1].Cells.Add(new TableCell(new Paragraph(new Run(cell))));}trg.Rows.Add(new TableRow());TableRow totalRow=trg.Rows[2];totalRow.Cells.Add(new TableCell(new Paragraph(new Run("ИТОГО"))){ColumnSpan=3});totalRow.Cells[0].FontWeight=FontWeights.Bold;totalRow.Cells.Add(new TableCell(new Paragraph(new Run($"{SelectContract?.Summ}"))){FontWeight=FontWeights.Bold});totalRow.Cells.Add(new TableCell(new Paragraph(new Run(""))));totalRow.Cells.Add(new TableCell(new Paragraph(new Run($"{SelectContract?.Summ \* (nds / 100)}"))));totalRow.Cells.Add(new TableCell(new Paragraph(new Run($"{SelectContract?.Summ \* (1 + nds / 100)}"))));section.Blocks.Add(table);Paragraph totals=new();totals.Inlines.Add(new Bold(new Run("Сумма НДС: ")));totals.Inlines.Add(new Run($"{SelectContract?.Summ \* (nds / 100)}:N2"+Properties.Settings.Default.Currency));totals.Inlines.Add(new LineBreak());totals.Inlines.Add(new Bold(new Run("Всего с НДС: ")));totals.Inlines.Add(new Run($"{SelectContract?.Summ \* (1 + nds / 100)}:N2"+Properties.Settings.Default.Currency));totals.Margin=new Thickness(0,10,0,0);section.Blocks.Add(totals);Table infoTable=new(){CellSpacing=0,Margin=new Thickness(20,20,0,0)};infoTable.Columns.Add(new TableColumn{Width=new GridLength(1,GridUnitType.Star)});infoTable.Columns.Add(new TableColumn{Width=new GridLength(1,GridUnitType.Star)});TableRowGroup infoGroup=new();infoTable.RowGroups.Add(infoGroup);TableRow row=new TableRow();infoGroup.Rows.Add(row);Paragraph buyer=new Paragraph();buyer.Inlines.Add(new Bold(new Run("Покупатель")));buyer.Inlines.Add(new LineBreak());buyer.Inlines.Add(new Run($"{SelectContract?.Counterpartie?.Name}\n"+$"{SelectContract?.Counterpartie?.LegalAddress}\n"+$"БИК: {SelectContract?.Counterpartie?.BankAccount}\n"+$"УНП: {SelectContract?.Counterpartie?.Unp}\n"+$"Тел: {SelectContract?.Counterpartie?.PhoneNomber}"));row.Cells.Add(new TableCell(buyer){BorderThickness=new Thickness(0)});Paragraph supplier=new Paragraph();supplier.Inlines.Add(new Bold(new Run("Поставщик")));supplier.Inlines.Add(new LineBreak());supplier.Inlines.Add(new Run($"ООО «Вактайм»\n"+$"231000 г. Сморгонь, Гродненская обл., ул. Пушкина 91\n"+$"p/c BY20AKBB30120877400144200000\n"+$"БИК АКВВВУ2Х в ЦБУ 423 ОЛО АСБ «Беларусбанк» г. Сморгонь ул. Советская 36\n"+$"УНН: 590976294\n"+$"ΟКПΟ: 300438284000\n"+$"Тел. GSM: +375 29-6299854;\n"+$"Факс: +375(1592)-4-55-23"));row.Cells.Add(new TableCell(supplier){BorderThickness=new Thickness(0)});section.Blocks.Add(infoTable);}}}<Page x:Class="VacTrack.ViewReport.EmployeeDivisionReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="EmployeeDivisionReport"><Page.DataContext><local:EmployeeDivisionReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><materialDesign:PopupBox StaysOpen="True" Margin="20,0,0,0" ToolTip="Выбор типа групировки" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FormatListGroupPlus" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Margin="0,5,0,0"><RadioButton Content="Групировка по должности" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByPost}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Групировка по подразделению" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByDivision}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Без групировки" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeNoGrouped}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/></StackPanel></materialDesign:PopupBox><materialDesign:PopupBox StaysOpen="True" Margin="10,0,0,0" ToolTip="Настиройки фильтрации" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FilterSettingsOutline" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Width="200" Background="{DynamicResource MaterialDesign.Brush.Background}"><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding EmployPosts}" SelectedItem="{Binding FilterByPost, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по должности" materialDesign:TextFieldAssist.HasClearButton="True"/><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding EmployDivisions}" SelectedItem="{Binding FilterByDivision, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по подразделению" materialDesign:TextFieldAssist.HasClearButton="True"/><Label Content="За период" Margin="0,15,0,0"/><DatePicker SelectedDate="{Binding FilterStartDate, UpdateSourceTrigger=PropertyChanged}" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="С" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><DatePicker SelectedDate="{Binding FilterEndDate, UpdateSourceTrigger=PropertyChanged}" Margin="0,5,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="По" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><Button Content="Очистить фильтр" Margin="0,15,0,0" Style="{StaticResource MaterialDesignOutlinedButton}" Command="{Binding ClearFilterCommand}"/></StackPanel></materialDesign:PopupBox><materialDesign:PopupBox StaysOpen="True" Margin="10,0,0,0" ToolTip="Выбор необходимых колонок" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FormatColumns" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Margin="0,5,0,0"><ItemsControl ItemsSource="{Binding Columns}"><ItemsControl.ItemTemplate><DataTemplate><CheckBox Content="{Binding Name}" IsChecked="{Binding IsVisible, Mode=TwoWay}" Style="{StaticResource MaterialDesignFilterChipPrimaryCheckBox}" Width="150" Margin="1" Click="CheckBox\_Click"/></DataTemplate></ItemsControl.ItemTemplate></ItemsControl></StackPanel></materialDesign:PopupBox><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class EmployeeDivisionReport : Page,ICachedPage{private EmployeeDivisionReportViewModel ThisViewModel=>(EmployeeDivisionReportViewModel)DataContext;public EmployeeDivisionReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();private void CheckBox\_Click(object sender,RoutedEventArgs e)=>ThisViewModel.OpenFromCache();}public enum GroupedType{NoGrouped,GroupedByDivision,GroupedByPost}public class ColumnSetting : INotifyPropertyChanged{private bool \_isVisible;public bool IsVisible{get=>\_isVisible;set{\_isVisible=value;OnPropertyChanged();}}public required string Name{get;set;}public required string Code{get;set;}public event PropertyChangedEventHandler? PropertyChanged;protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));}public class EmployeeDivisionReportViewModel : BaseReportViewModel<Employee>{public ObservableCollection<Division>? EmployDivisions{get;set;}public ObservableCollection<Post>? EmployPosts{get;set;}#region properties private GroupedType GroupedType=GroupedType.NoGrouped;public bool IsGroupedTypeNoGrouped{get=>GroupedType==GroupedType.NoGrouped;set{if(value){GroupedType=GroupedType.NoGrouped;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}public bool IsGroupedTypeGroupedByDivision{get=>GroupedType==GroupedType.GroupedByDivision;set{if(value){GroupedType=GroupedType.GroupedByDivision;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByDivision));}}}public bool IsGroupedTypeGroupedByPost{get=>GroupedType==GroupedType.GroupedByPost;set{if(value){GroupedType=GroupedType.GroupedByPost;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByPost));}}}private Post? \_FilterByPost;public Post? FilterByPost{get=>\_FilterByPost;set{SetProperty(ref \_FilterByPost,value);LoadData();}}private Division? \_FilterByDivision;public Division? FilterByDivision{get=>\_FilterByDivision;set{SetProperty(ref \_FilterByDivision,value);LoadData();}}private DateTime? \_FilterStartDate;public DateTime? FilterStartDate{get=>\_FilterStartDate;set{SetProperty(ref \_FilterStartDate,value);LoadData();}}private DateTime? \_FilterEndDate;public DateTime? FilterEndDate{get=>\_FilterEndDate;set{SetProperty(ref \_FilterEndDate,value);LoadData();}}public ObservableCollection<ColumnSetting>Columns{get;set;}=[ new ColumnSetting{Name="ФИО",Code="fio",IsVisible=true},new ColumnSetting{Name="Подразделение",Code="division",IsVisible=true},new ColumnSetting{Name="Должность",Code="post",IsVisible=true},new ColumnSetting{Name="Дата приема",Code="dateHire",IsVisible=true},new ColumnSetting{Name="Дата рождения",Code="dateOfBirth",IsVisible=true},new ColumnSetting{Name="Телефон",Code="phoneNumber",IsVisible=true},new ColumnSetting{Name="Адресс",Code="address",IsVisible=true},new ColumnSetting{Name="Оклад,\n"+Properties.Settings.Default.Currency,Code="salary",IsVisible=true},new ColumnSetting{Name="Дата увольнения",Code="dateDismissal",IsVisible=false},];#endregion public ICommand ClearFilterCommand{get;}public EmployeeDivisionReportViewModel(){ClearFilterCommand=new RelayCommand(ClearFilter);}protected override void LoadData(){EmployDivisions=new ObservableCollection<Division>([..Db.Divisions]);EmployPosts=new ObservableCollection<Post>([..Db.Posts]);DbSet=Db.Set<Employee>();DbSet.Include(e=>e.Division).Include(e=>e.Post).Load();Items=[..DbSet.Local.Where(item=>(FilterByPost==null||item.Post?.Id==FilterByPost.Id)&&(FilterByDivision==null||item.Division?.Id==FilterByDivision.Id)&&((FilterStartDate==null&&FilterEndDate==null)||(FilterStartDate!=null&&FilterEndDate!=null&&((item.DateHire>=FilterStartDate&&item.DateHire<=FilterEndDate)||(item.DateDismissal>=FilterStartDate&&item.DateDismissal<=FilterEndDate)))||(FilterStartDate!=null&&FilterEndDate==null&&(item.DateHire==FilterStartDate||item.DateDismissal==FilterStartDate))||(FilterStartDate==null&&FilterEndDate!=null&&(item.DateHire==FilterEndDate||item.DateDismissal==FilterEndDate)))).ToList()];}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();switch(GroupedType){case GroupedType.GroupedByDivision: CreateGroupedByDivisionRows(ref dataGroup);break;case GroupedType.GroupedByPost: CreateGroupedByPostRows(ref dataGroup);break;case GroupedType.NoGrouped: CreateNoGroupedRows(ref dataGroup);break;}table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);doc.Blocks.Add(new Paragraph(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private List<string>GetVisibleValues(Employee item,string codeEmptyCol=""){var result=new List<string>();foreach(var column in Columns){if(!column.IsVisible)continue;if(column.Code==codeEmptyCol){result.Add(string.Empty);continue;}switch(column.Code){case "fio": result.Add($"{item.Fio}");break;case "division": result.Add($"{item.Division?.Name}");break;case "post": result.Add($"{item.Post?.Name}");break;case "dateHire": result.Add($"{item.DateHire:dd.MM.yyyy}");break;case "dateOfBirth": result.Add($"{item.DateOfBirth:dd.MM.yyyy}");break;case "phoneNumber": result.Add($"{item.PhoneNumber}");break;case "address": result.Add($"{item.Address}");break;case "salary": result.Add($"{item.Salary:N2}");break;case "dateDismissal": result.Add($"{item.DateDismissal:dd.MM.yyyy}");break;default: result.Add(string.Empty);break;}}return result;}private void CreateGroupedByDivisionRows(ref TableRowGroup dataGroup){var groupedItems=Items.GroupBy(item=>item.Division?.Name);foreach(var group in groupedItems){var headerValues=Columns.Where(c=>c.IsVisible).Select(c=>c.Code=="division" ? group.Key : "").ToList();#pragma warning disable CS8620 dataGroup.Rows.Add(CreateRow(headerValues));#pragma warning restore CS8620 foreach(var item in group){var visibleValues=GetVisibleValues(item,"division");dataGroup.Rows.Add(CreateRow(visibleValues));}}}private void CreateGroupedByPostRows(ref TableRowGroup dataGroup){var groupedItems=Items.GroupBy(item=>item.Post?.Name);foreach(var group in groupedItems){var headerValues=Columns.Where(c=>c.IsVisible).Select(c=>c.Code=="post" ? group.Key : "").ToList();#pragma warning disable CS8620 dataGroup.Rows.Add(CreateRow(headerValues));#pragma warning restore CS8620 foreach(var item in group){var visibleValues=GetVisibleValues(item,"post");dataGroup.Rows.Add(CreateRow(visibleValues));}}}private void CreateNoGroupedRows(ref TableRowGroup dataGroup){foreach(var item in Items)dataGroup.Rows.Add(CreateRow(GetVisibleValues(item)));}private void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();foreach(var col in Columns)if(col.IsVisible)headerRow.Cells.Add(new TableCell(new Paragraph(new Run(col.Name))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Отчет по сотрудникам")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filterByPost=FilterByPost==null ? string.Empty : $"\nПо должности: {FilterByPost.Name}";string filterByDivision=FilterByDivision==null ? string.Empty : $"\nПо подразделению: {FilterByDivision.Name}";string periodFilter=(FilterStartDate,FilterEndDate)switch{(not null,not null)=>$"\nЗа период с {FilterStartDate:dd.MM.yyyy} по {FilterEndDate:dd.MM.yyyy}",(not null,null)=>$"\nНа день {FilterStartDate:dd.MM.yyyy}",(null,not null)=>$"\nНа день {FilterEndDate:dd.MM.yyyy}",\_=>string.Empty};string filteredText=filterByPost+filterByDivision+periodFilter;string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}private void ClearFilter(object obj){FilterByDivision=null;FilterByPost=null;FilterStartDate=null;FilterEndDate=null;}}}<Page x:Class="VacTrack.ViewReport.MaterialUsageReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="MaterialUsageReport"><Page.DataContext><local:MaterialUsageReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><ListBox Style="{StaticResource MaterialDesignFilterChipPrimaryListBox}" Height="46" Margin="20,0,0,0"><ListBoxItem Content="Группировка" IsSelected="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Итого группы" IsSelected="{Binding IsGroupTotalEnabled}" IsEnabled="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Общие итоги" IsSelected="{Binding AreOverallTotalsEnabled}"/></ListBox><ComboBox ItemsSource="{Binding Products}" Style="{StaticResource MaterialDesignFloatingHintComboBox}" Width="130" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" SelectedItem="{Binding SelectedProduct}" Margin="20,0,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по изделию" materialDesign:TextFieldAssist.HasClearButton="True"/><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class MaterialUsageReport : Page,ICachedPage{private MaterialUsageReportViewModel ThisViewModel=>(MaterialUsageReportViewModel)DataContext;public MaterialUsageReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class MaterialUsageReportViewModel : BaseReportViewModel<Product\_Material>{public List<Product>? Products{get;set;}#region properties private bool \_IsGroupingEnabled=true;public bool IsGroupingEnabled{get=>\_IsGroupingEnabled;set{SetProperty(ref \_IsGroupingEnabled,value);Refresh(null);}}private bool \_IsGroupTotalEnabled=true;public bool IsGroupTotalEnabled{get=>\_IsGroupTotalEnabled;set{SetProperty(ref \_IsGroupTotalEnabled,value);Refresh(null);}}private bool \_AreOverallTotalsEnabled=true;public bool AreOverallTotalsEnabled{get=>\_AreOverallTotalsEnabled;set{SetProperty(ref \_AreOverallTotalsEnabled,value);Refresh(null);}}private Product? \_SelectedProduct;public Product? SelectedProduct{get=>\_SelectedProduct;set{SetProperty(ref \_SelectedProduct,value);LoadData();}}#endregion protected override void LoadData(){Products=new List<Product>([..Db.Products]);DbSet=Db.Set<Product\_Material>();DbSet.Include(e=>e.Material).ThenInclude(c=>c!=null ? c.Unit : null).Include(e=>e.Product).Load();if(SelectedProduct==null)Items=DbSet.Local.ToObservableCollection();else Items=[..DbSet.Local.Where(item=>item.Product?.Id==SelectedProduct.Id).ToList()];}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc,SelectedProduct);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSumm=0;if(IsGroupingEnabled)CreateGroupedRows(ref dataGroup,ref totalSumm);else CreateUngroupedRows(ref dataGroup,ref totalSumm);if(AreOverallTotalsEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","",$"{totalSumm:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);doc.Blocks.Add(new Paragraph(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private void CreateGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Product?.Name,item=>item.GetSum,key=>[$"{key}","","","",""],total=>["Итого","","","",$"{total:N2}"],item=>[ String.Empty,$"{item.Material?.Name}",$"{item.Quantity} {item.Material?.Unit?.Name}",$"{item.Material?.Price:N2}",$"{item.GetSum:N2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateUngroupedRows(ref TableRowGroup dataGroup,ref double totalSumm){foreach(var item in Items){double price=item.GetSum;totalSumm+=price;dataGroup.Rows.Add(CreateRow([ $"{item.Product?.Name}",$"{item.Material?.Name}",$"{item.Quantity} {item.Material?.Unit?.Name}",$"{item.Material?.Price:N2}",$"{price:N2}" ]));}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Изделие"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Материал"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Стоимость,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private static void AddHeader(ref FlowDocument doc,Product? SelectedProduct){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Отчет по использованию материалов на изделия")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filteredText=SelectedProduct==null ? "" : $"\nДля изделия: {SelectedProduct.Name}";string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}}}<Page x:Class="VacTrack.ViewReport.ProcurementSheetReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="ProcurementSheetReport"><Page.DataContext><local:ProcurementSheetReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><ListBox Style="{StaticResource MaterialDesignFilterChipPrimaryListBox}" Height="46" Margin="20,0,0,0"><ListBoxItem Content="Группировка" IsSelected="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Итого группы" IsSelected="{Binding IsGroupTotalEnabled}" IsEnabled="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Общие итоги" IsSelected="{Binding AreOverallTotalsEnabled}"/></ListBox><ComboBox ItemsSource="{Binding Products}" Style="{StaticResource MaterialDesignFloatingHintComboBox}" Width="130" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" SelectedItem="{Binding SelectedProduct}" Margin="20,0,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по изделию" materialDesign:TextFieldAssist.HasClearButton="True"/><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class ProcurementSheetReport : Page,ICachedPage{private ProcurementSheetReportViewModel ThisViewModel=>(ProcurementSheetReportViewModel)DataContext;public ProcurementSheetReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class ProcurementSheetReportViewModel : BaseReportViewModel<Product\_Material>{public required List<Product>Products{get;set;}public required List<Material>Materials{get;set;}private List<Material>TotalBuyMaterials{get;set;}=[];#region properties private bool \_IsGroupingEnabled=true;public bool IsGroupingEnabled{get=>\_IsGroupingEnabled;set{SetProperty(ref \_IsGroupingEnabled,value);Refresh(null);}}private bool \_IsGroupTotalEnabled=true;public bool IsGroupTotalEnabled{get=>\_IsGroupTotalEnabled;set{SetProperty(ref \_IsGroupTotalEnabled,value);Refresh(null);}}private bool \_AreOverallTotalsEnabled=true;public bool AreOverallTotalsEnabled{get=>\_AreOverallTotalsEnabled;set{SetProperty(ref \_AreOverallTotalsEnabled,value);Refresh(null);}}private Product? \_SelectedProduct;public Product? SelectedProduct{get=>\_SelectedProduct;set{SetProperty(ref \_SelectedProduct,value);LoadData();}}#endregion protected override void LoadData(){Products=new List<Product>([..Db.Products]);Materials=new List<Material>([..Db.Materials]);TotalBuyMaterials.Clear();DbSet=Db.Set<Product\_Material>();DbSet.Include(e=>e.Material).ThenInclude(c=>c!=null ? c.Unit : null).Include(e=>e.Product).Load();if(SelectedProduct==null)Items=DbSet.Local.ToObservableCollection();else Items=[..DbSet.Local.Where(item=>item.Product?.Id==SelectedProduct.Id).ToList()];}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc,SelectedProduct);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSumm=0;if(IsGroupingEnabled)CreateGroupedRows(ref dataGroup,ref totalSumm);else CreateUngroupedRows(ref dataGroup,ref totalSumm);if(AreOverallTotalsEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","","","",$"{totalSumm:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);doc.Blocks.Add(new Paragraph(new Run("Итого нужно:")){FontSize=12,Margin=new Thickness(0,0,0,10)});doc.Blocks.Add(CreatePurchaseTable(TotalBuyMaterials));Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);doc.Blocks.Add(new Paragraph(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private static Table CreatePurchaseTable(IEnumerable<Material>materials){var table=new Table{CellSpacing=0,Margin=new Thickness(5),TextAlignment=TextAlignment.Left,FontSize=12};table.Columns.Add(new TableColumn{Width=new GridLength(3,GridUnitType.Star)});table.Columns.Add(new TableColumn{Width=new GridLength(1,GridUnitType.Star)});table.Columns.Add(new TableColumn{Width=new GridLength(2,GridUnitType.Star)});table.Columns.Add(new TableColumn{Width=new GridLength(2,GridUnitType.Star)});var rowGroup=new TableRowGroup();var headerRow=CreateRow([ "Наименование","Количество","Цена за ед.,\n"+Properties.Settings.Default.Currency,"Сумма,\n"+Properties.Settings.Default.Currency ]);headerRow.FontWeight=FontWeights.Bold;rowGroup.Rows.Add(headerRow);foreach(var m in materials){var count=$"{m.Count:N0} {m.Unit?.Name ?? string.Empty}".Trim();var price=$"{m.Price:N2}";var sum=$"{m.GetSum:N2}";rowGroup.Rows.Add(CreateRow([ m.Name,count,price,sum ]));}var totalRow=CreateRow([ "Итого:",string.Empty,string.Empty,$"{materials.Sum(m => m.GetSum):N2}" ]);totalRow.FontWeight=FontWeights.SemiBold;rowGroup.Rows.Add(totalRow);table.RowGroups.Add(rowGroup);return table;}private void CreateGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){Dictionary<int,double>materialDictionary=Materials.ToDictionary(m=>m.Id,m=>m.Count);var groupedItems=Items.GroupBy(item=>item.Product?.Name);foreach(var group in groupedItems){dataGroup.Rows.Add(CreateRow([$"{group.Key}","","","","","",""]));double summ=0;foreach(var item in group){var unit=item.Material?.Unit?.Name ?? "";var available=materialDictionary[item.MaterialId];var shortage=GetShortage(item.Quantity,available);var cost=CalculateCost(item.Material?.Price,shortage);dataGroup.Rows.Add(CreateRow([ string.Empty,$"{item.Material?.Name}",$"{item.Quantity} {unit}",$"{available:N0} {unit}",$"{shortage:N0} {unit}",$"{item.Material?.Price:N2}",$"{cost:N2}" ]));summ+=cost;materialDictionary[item.MaterialId]=Math.Max(0,materialDictionary.GetValueOrDefault(item.MaterialId)-item.Quantity);if(shortage>0&&item.Material!=null)AddOrUpdateMaterial(item.Material,shortage);}if(IsGroupTotalEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","","","",$"{summ:N2}"]));totalSum+=summ;}}private void CreateUngroupedRows(ref TableRowGroup dataGroup,ref double totalSumm){Dictionary<int,double>materialDictionary=Materials.ToDictionary(m=>m.Id,m=>m.Count);foreach(var item in Items){var unit=item.Material?.Unit?.Name ?? "";var available=materialDictionary[item.MaterialId];var shortage=GetShortage(item.Quantity,available);var cost=CalculateCost(item.Material?.Price,shortage);totalSumm+=cost;dataGroup.Rows.Add(CreateRow([ $"{item.Product?.Name}",$"{item.Material?.Name}",$"{item.Quantity} {unit}",$"{available:N0} {unit}",$"{shortage:N0} {unit}",$"{item.Material?.Price:N2}",$"{cost:N2}" ]));materialDictionary[item.MaterialId]=Math.Max(0,materialDictionary.GetValueOrDefault(item.MaterialId)-item.Quantity);if(shortage>0&&item.Material!=null)AddOrUpdateMaterial(item.Material,shortage);}}private static double GetShortage(double? required,double? available)=>Math.Max(0,(required ?? 0)-(available ?? 0));private static double CalculateCost(double? price,double? count)=>(price ?? 0)\*(count ?? 0);private void AddOrUpdateMaterial(Material newMaterial,double shortage){var existingMaterial=TotalBuyMaterials.FirstOrDefault(m=>m.Id==newMaterial.Id);if(existingMaterial!=null){existingMaterial.Count+=shortage;}else{TotalBuyMaterials.Add(new Material{Id=newMaterial.Id,Name=newMaterial.Name,UnitId=newMaterial.UnitId,Unit=newMaterial.Unit,LocationId=newMaterial.LocationId,Location=newMaterial.Location,Count=shortage,Price=newMaterial.Price,ProductMaterials=newMaterial.ProductMaterials,Receipts=newMaterial.Receipts,});}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Изделие"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("материал"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("необходимо"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Есть на складе"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Нехватет"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Стоимость,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private static void AddHeader(ref FlowDocument doc,Product? SelectedProduct){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Список закупок")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filteredText=SelectedProduct==null ? "" : $"\nДля изделия: {SelectedProduct.Name}";string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}}}<Page x:Class="VacTrack.ViewReport.ProductSalesReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:converter="clr-namespace:VacTrack.Converter" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="ProductSalesReport"><Page.DataContext><local:ProductSalesReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Page.Resources><converter:InvertBooleanConverter x:Key="InvertBooleanConverter"/></Page.Resources><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><materialDesign:PopupBox StaysOpen="True" Margin="20,0,0,0" ToolTip="Выбор типа групировки" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FormatListGroupPlus" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Margin="0,5,0,0"><RadioButton Content="Групировка по контрагенту" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByCounterpartie}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Групировка по изделию" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeGroupedByProduct}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="Без групировки" GroupName="GroupType" IsChecked="{Binding IsGroupedTypeNoGrouped}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><Border Margin="0,15,0,0" Background="{DynamicResource MaterialDesign.Brush.Background}" CornerRadius="15"><StackPanel><Label Content="Настройка итогов"/><ListBox Style="{StaticResource MaterialDesignFilterChipPrimaryListBox}" Height="46"><ListBoxItem Content="Итого группы" IsSelected="{Binding IsGroupTotalEnabled}" IsEnabled="{Binding IsGroupedTypeNoGrouped, Converter={StaticResource InvertBooleanConverter}, UpdateSourceTrigger=PropertyChanged}"/><ListBoxItem Content="Общие итоги" IsSelected="{Binding AreOverallTotalsEnabled}"/></ListBox></StackPanel></Border></StackPanel></materialDesign:PopupBox><materialDesign:PopupBox StaysOpen="True" Margin="10,0,0,0" ToolTip="Настиройки фильтрации" Style="{StaticResource MaterialDesignMultiFloatingActionPopupBox}" Width="30" Height="30" PlacementMode="BottomAndAlignLeftEdges"><materialDesign:PopupBox.ToggleContent><materialDesign:PackIcon Kind="FilterSettingsOutline" Width="30" Foreground="{DynamicResource MaterialDesignBody}"/></materialDesign:PopupBox.ToggleContent><StackPanel Orientation="Vertical" Width="200" Background="{DynamicResource MaterialDesign.Brush.Background}"><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding ContractCounterpartie}" SelectedItem="{Binding FilterByCounterpartie, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по котрагенту" materialDesign:TextFieldAssist.HasClearButton="True"/><ComboBox Margin="0,10,0,0" Style="{StaticResource MaterialDesignOutlinedComboBox}" ItemsSource="{Binding ContractProduct}" SelectedItem="{Binding FilterByProduct, UpdateSourceTrigger=PropertyChanged}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по изделию" materialDesign:TextFieldAssist.HasClearButton="True"/><Label Content="За период" Margin="0,15,0,0"/><DatePicker SelectedDate="{Binding FilterStartDate, UpdateSourceTrigger=PropertyChanged}" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="С" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><DatePicker SelectedDate="{Binding FilterEndDate, UpdateSourceTrigger=PropertyChanged}" Margin="0,5,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="По" Style="{StaticResource MaterialDesignOutlinedDatePicker}"/><Button Content="Очистить фильтр" Margin="0,15,0,0" Style="{StaticResource MaterialDesignOutlinedButton}" Command="{Binding ClearFilterCommand}"/></StackPanel></materialDesign:PopupBox><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class ProductSalesReport : Page{private ProductSalesReportViewModel ThisViewModel=>(ProductSalesReportViewModel)DataContext;public ProductSalesReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public enum SaleGroupedType{NoGrouped,GroupedByCounterpartie,GroupedByProduct}public class ProductSalesReportViewModel : BaseReportViewModel<Sale>{public ObservableCollection<Counterpartie>? ContractCounterpartie{get;set;}public ObservableCollection<Product>? ContractProduct{get;set;}#region properties private SaleGroupedType GroupedType=SaleGroupedType.NoGrouped;public bool IsGroupedTypeNoGrouped{get=>GroupedType==SaleGroupedType.NoGrouped;set{if(value){GroupedType=SaleGroupedType.NoGrouped;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}public bool IsGroupedTypeGroupedByCounterpartie{get=>GroupedType==SaleGroupedType.GroupedByCounterpartie;set{if(value){GroupedType=SaleGroupedType.GroupedByCounterpartie;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByCounterpartie));OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}public bool IsGroupedTypeGroupedByProduct{get=>GroupedType==SaleGroupedType.GroupedByProduct;set{if(value){GroupedType=SaleGroupedType.GroupedByProduct;Refresh(null);OnPropertyChanged(nameof(IsGroupedTypeGroupedByProduct));OnPropertyChanged(nameof(IsGroupedTypeNoGrouped));}}}private bool \_IsGroupTotalEnabled=true;public bool IsGroupTotalEnabled{get=>\_IsGroupTotalEnabled;set{SetProperty(ref \_IsGroupTotalEnabled,value);Refresh(null);}}private bool \_AreOverallTotalsEnabled=true;public bool AreOverallTotalsEnabled{get=>\_AreOverallTotalsEnabled;set{SetProperty(ref \_AreOverallTotalsEnabled,value);Refresh(null);}}private Counterpartie? \_FilterByCounterpartie;public Counterpartie? FilterByCounterpartie{get=>\_FilterByCounterpartie;set{SetProperty(ref \_FilterByCounterpartie,value);LoadData();}}private Product? \_FilterByProduct;public Product? FilterByProduct{get=>\_FilterByProduct;set{SetProperty(ref \_FilterByProduct,value);LoadData();}}private DateTime? \_FilterStartDate;public DateTime? FilterStartDate{get=>\_FilterStartDate;set{SetProperty(ref \_FilterStartDate,value);LoadData();}}private DateTime? \_FilterEndDate;public DateTime? FilterEndDate{get=>\_FilterEndDate;set{SetProperty(ref \_FilterEndDate,value);LoadData();}}#endregion public System.Windows.Input.ICommand ClearFilterCommand{get;}public ProductSalesReportViewModel(){ClearFilterCommand=new RelayCommand(ClearFilter);}protected override void LoadData(){ContractCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);ContractProduct=new ObservableCollection<Product>([..Db.Products]);DbSet=Db.Set<Sale>();DbSet.Include(e=>e.Contract).Include(c=>c.Contract!=null ? c.Contract.Product : null).ThenInclude(e=>e!=null ? e.Unit : null).Include(c=>c.Contract!=null ? c.Contract.Counterpartie : null).Load();Items=new ObservableCollection<Sale>(DbSet.Local.Where(item=>(FilterByCounterpartie==null||item.Contract?.Counterpartie?.Id==FilterByCounterpartie.Id)&&(FilterByProduct==null||item.Contract?.Product?.Id==FilterByProduct.Id)&&((FilterStartDate==null&&FilterEndDate==null)||(FilterStartDate!=null&&FilterEndDate!=null&&item.Date>=FilterStartDate&&item.Date<=FilterEndDate)||(FilterStartDate!=null&&FilterEndDate==null&&item.Date==FilterStartDate)||(FilterStartDate==null&&FilterEndDate!=null&&item.Date==FilterEndDate))).ToList());}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSum=0;switch(GroupedType){case SaleGroupedType.GroupedByCounterpartie: CreateGroupedByCounterpartie(ref dataGroup,ref totalSum);break;case SaleGroupedType.GroupedByProduct: CreateGroupedByProduct(ref dataGroup,ref totalSum);break;case SaleGroupedType.NoGrouped: CreateNoGroupedRows(ref dataGroup,ref totalSum);break;}if(AreOverallTotalsEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","",$"{totalSum:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleAccountant);doc.Blocks.Add(new Paragraph(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private void CreateGroupedByCounterpartie(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Contract?.Counterpartie?.Name,item=>item.Summ,key=>["",$"{key}","","",""],total=>["Итого","","","",$"{total:N2}"],item=>[ $"{item.Date:dd.MM.yyyy}",String.Empty,$"{item.Contract?.Product?.Name}",$"{item.Count} {item.Contract?.Product?.Unit?.Name}",$"{item.Summ:N2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateGroupedByProduct(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Contract?.Product?.Name,item=>item.Summ,item=>item.Count,key=>["","",$"{key}","","",""],cntTotal=>["Итого","","",$"{cntTotal.Item1}",$"{cntTotal.Item2:N2}"],item=>[ $"{item.Date:dd.MM.yyyy}",$"{item.Contract?.Counterpartie?.Name}",String.Empty,$"{item.Count} {item.Contract?.Product?.Unit?.Name}",$"{item.Summ:N2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateNoGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){foreach(var item in Items){dataGroup.Rows.Add(CreateRow([ $"{item.Date:dd.MM.yyyy}",$"{item.Contract?.Counterpartie?.Name}",$"{item.Contract?.Product?.Name}",$"{item.Count} {item.Contract?.Product?.Unit?.Name}",$"{item.Summ:N2}" ]));totalSum+=item.Summ;}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Дата реализации"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Контрагент"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Изделие"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Отчет по реализации продукции")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filterByPost=FilterByCounterpartie==null ? string.Empty : $"\nПо контрагенту: {FilterByCounterpartie.Name}";string filterByDivision=FilterByProduct==null ? string.Empty : $"\nПо изделию: {FilterByProduct.Name}";string periodFilter=(FilterStartDate,FilterEndDate)switch{(not null,not null)=>$"\nЗа период с {FilterStartDate:dd.MM.yyyy} по {FilterEndDate:dd.MM.yyyy}",(not null,null)=>$"\nНа день {FilterStartDate:dd.MM.yyyy}",(null,not null)=>$"\nНа день {FilterEndDate:dd.MM.yyyy}",\_=>string.Empty};string filteredText=filterByPost+filterByDivision+periodFilter;string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}private void ClearFilter(object obj){FilterByCounterpartie=null;FilterByProduct=null;FilterStartDate=null;FilterEndDate=null;}}}<Window x:Class="VacTrack.ViewReport.ReceiptReport" 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:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" mc:Ignorable="d" Title="Печать накладной" Height="450" Width="800"><Window.DataContext><local:ReceiptReportViewModel/></Window.DataContext><Window.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Window.InputBindings><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Window>namespace VacTrack.ViewReport{public partial class ReceiptReport : Window{private ReceiptReportViewModel ThisViewModel=>(ReceiptReportViewModel)DataContext;public ReceiptReport(ObservableCollection<Receipt>items,DialogWindows.ReceiptsPrintVM receiptsPrintVM){InitializeComponent();ThisViewModel.FilteredItems=items;ThisViewModel.SelectCounterpartie=receiptsPrintVM.SelectCounterpartie;ThisViewModel.SelectLocation=receiptsPrintVM.SelectLocation;ThisViewModel.SelectMaterial=receiptsPrintVM.SelectMaterial;ThisViewModel.SelectStartDate=receiptsPrintVM.SelectStartDate;ThisViewModel.SelectEndDate=receiptsPrintVM.SelectEndDate;}}public class ReceiptReportViewModel : BaseReportViewModel<Receipt>{#region properties private Counterpartie? \_selectCounterpartie;public Counterpartie? SelectCounterpartie{get=>\_selectCounterpartie;set{if(\_selectCounterpartie==value)return;SetProperty(ref \_selectCounterpartie,value);Refresh(null);}}private Location? \_selectLocation;public Location? SelectLocation{get=>\_selectLocation;set{Location? location=value!=null ? Db.Locations.Include(e=>e.Employee).FirstOrDefault(e=>e.Id==value.Id): null;if(\_selectLocation==location)return;SetProperty(ref \_selectLocation,location);Refresh(null);}}private Material? \_selectMaterial;public Material? SelectMaterial{get=>\_selectMaterial;set{if(\_selectMaterial==value)return;SetProperty(ref \_selectMaterial,value);Refresh(null);}}private DateTime? \_selectStartDate;public DateTime? SelectStartDate{get=>\_selectStartDate;set{if(\_selectStartDate==value)return;SetProperty(ref \_selectStartDate,value);Refresh(null);}}private DateTime? \_selectEndDate;public DateTime? SelectEndDate{get=>\_selectEndDate;set{if(\_selectEndDate==value)return;SetProperty(ref \_selectEndDate,value);Refresh(null);}}private ObservableCollection<Receipt>? \_filteredItems;public ObservableCollection<Receipt>? FilteredItems{get=>\_filteredItems;set=>SetProperty(ref \_filteredItems,value);}#endregion public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSum=0;CreateNoGroupedRows(ref dataGroup,ref totalSum);dataGroup.Rows.Add(CreateRow(["Итого","X","X","X",$"{totalSum:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=SelectLocation==null ? Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleStorekeeper): Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==SelectLocation.EmployeeId);Paragraph headerParagraph=new(new Run($"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);return doc;}private void CreateNoGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){if(FilteredItems==null)return;foreach(Receipt item in FilteredItems){dataGroup.Rows.Add(CreateRow([ $"{item.Material?.Name}",$"{item.Count}",$"{item.Material?.Unit?.Name}",$"{item.Material?.Price}",$"{item.Summ:N2}" ]));totalSum+=item.Summ;}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("ТМЦ"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("ед.изм."))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("цена,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){string? startDate=SelectStartDate?.ToString("d MMMM yyyy 'г.'",new CultureInfo("ru-RU"));string? endDate=SelectEndDate?.ToString("d MMMM yyyy 'г.'",new CultureInfo("ru-RU"));string date=startDate!=null&&endDate!=null ? $"За период с {startDate} по {endDate}" : startDate!=null ? $"От {startDate}" : $"От {endDate}";Paragraph title=new(new Run($"Приходная накладная № {FilteredItems?[0]?.Id}\n"+date)){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string headerText=$"\nПоставщик: {SelectCounterpartie?.Name}"+$"\nСклад получатель: {SelectLocation?.Name}, {SelectLocation?.Employee?.Fio}";Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}}}<Window x:Class="VacTrack.ViewReport.SaleReport" 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:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" mc:Ignorable="d" Title="Товарная накладная" Height="450" Width="800"><Window.DataContext><local:SaleReportViewModel/></Window.DataContext><Window.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Window.InputBindings><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Window>namespace VacTrack.ViewReport{public partial class SaleReport : Window{private SaleReportViewModel ThisViewModel=>(SaleReportViewModel)DataContext;public SaleReport(Sale sale){InitializeComponent();ThisViewModel.SelectSale=sale;}}public class SaleReportViewModel : BaseReportViewModel<Sale>{private Sale? \_selectSale;public Sale? SelectSale{get=>\_selectSale;set{SetProperty(ref \_selectSale,value);Refresh(null);}}private readonly double \_nds=Properties.Settings.Default.Nds/100;public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();CreateNoGroupedRows(ref dataGroup);dataGroup.Rows.Add(CreateRow([ "Итого","X",$"{SelectSale?.Count}",$"{SelectSale?.Summ:N2}","X",$"{SelectSale?.Summ \* \_nds:F2}",$"{SelectSale?.Summ \* (1 + \_nds):N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? productReleaseApprover=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ProductReleaseApprover);Employee? productSubmitter=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ProductSubmitter);string footerText=$"Отпуск разрешил: {productReleaseApprover?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ {productReleaseApprover?.Fio}"+$"\nСдал грузоотправитель: {productSubmitter?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ {productSubmitter?.Fio}";Paragraph headerParagraph=new(new Run(footerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);return doc;}private void CreateNoGroupedRows(ref TableRowGroup dataGroup){dataGroup.Rows.Add(CreateRow([ $"{SelectSale?.Contract?.Product?.Name}",$"{SelectSale?.Contract?.Product?.Unit?.Name}",$"{SelectSale?.Count}",$"{SelectSale?.Summ:N2}",$"{\_nds \* 100:F2}",$"{SelectSale?.Summ \* \_nds:N2}",$"{SelectSale?.Summ \* (1 + \_nds):N2}" ]));}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Наименование товара"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Ед.изм."))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Цена,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Ставка НДС, %"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма НДС,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма с НДС,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){Paragraph title=new(new Run($"Товарная накладная №{SelectSale?.Id}\n"+$"От{SelectSale?.Date.ToString("d MMMM yyyy 'г.'", new CultureInfo("ru-RU"))}")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string headerText=$"\nГрузоотправитель: ООО «ВакТайм», адресс: {Properties.Settings.Default.OrganizationAdress}"+$"\nГрузополучатель: {SelectSale?.Contract?.Counterpartie?.Name}, адрес: {SelectSale?.Contract?.Counterpartie?.PostalAddress}"+$"\nОснование отпуска: договор {SelectSale?.Contract?.Name} от {SelectSale?.Contract?.Date.ToString("dd.MM.yyyy", new CultureInfo("ru-RU"))}";Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}}}<Page x:Class="VacTrack.ViewReport.SalesStatisticsReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:lvc="clr-namespace:LiveChartsCore.SkiaSharpView.WPF;assembly=LiveChartsCore.SkiaSharpView.WPF" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" xmlns:converter="clr-namespace:VacTrack.Converter" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="SalesStatisticsReport"><Page.DataContext><local:SalesStatisticsViewModel/></Page.DataContext><Page.Resources><converter:InvertBooleanConverter x:Key="InvertBooleanConverter"/></Page.Resources><Grid><StackPanel Orientation="Horizontal" HorizontalAlignment="Left" VerticalAlignment="Top"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button x:Name="ExportBtn" Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" ToolTip="Экспорт графика" Click="ExportBtn\_Click"><materialDesign:PackIcon Kind="FileExport"/></Button><RadioButton Margin="20,0,0,0" Content="По годам" GroupName="GroupType" IsChecked="{Binding IsGroupedByYear, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="По месяцам" GroupName="GroupType" IsChecked="{Binding IsGroupedByYear, Converter={StaticResource InvertBooleanConverter}}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Margin="10,0,0,0" Content="По количеству" GroupName="LineType" IsChecked="{Binding IsCountGraph, UpdateSourceTrigger=PropertyChanged}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/><RadioButton Content="По цене" GroupName="LineType" IsChecked="{Binding IsCountGraph, Converter={StaticResource InvertBooleanConverter}}" Style="{StaticResource MaterialDesignChoiceChipPrimaryRadioButton}"/></StackPanel><lvc:CartesianChart x:Name="chartControl" Series="{Binding Series, UpdateSourceTrigger=PropertyChanged}" XAxes="{Binding XAxes, UpdateSourceTrigger=PropertyChanged}" YAxes="{Binding YAxes, UpdateSourceTrigger=PropertyChanged}" LegendPosition="Bottom" ZoomMode="Both" Margin="0,40,0,0"></lvc:CartesianChart></Grid></Page>namespace VacTrack.ViewReport{public partial class SalesStatisticsReport : Page{public SalesStatisticsReport(){InitTheme();InitializeComponent();}private static void InitTheme(){if(new PaletteHelper().GetTheme().GetBaseTheme()==BaseTheme.Dark)LiveCharts.Configure(settings=>settings.AddDefaultMappers().AddSkiaSharp().AddDarkTheme());else LiveCharts.Configure(settings=>settings.AddDefaultMappers().AddSkiaSharp().AddLightTheme());}private void ExportBtn\_Click(object sender,RoutedEventArgs e){LiveCharts.Configure(settings=>settings.AddDefaultMappers().AddSkiaSharp().AddLightTheme());try{var dlg=new SaveFileDialog{Title="Сохранить график как...",Filter="JPEG Image (\*.jpg;\*.jpeg)|\*.jpg;\*.jpeg|"+"PNG Image (\*.png)|\*.png|"+"WEBP Image (\*.webp)|\*.webp|"+"All Files (\*.\*)|\*.\*",DefaultExt="png",AddExtension=true,FilterIndex=2};if(dlg.ShowDialog()!=true)return;var path=dlg.FileName;var ext=Path.GetExtension(path).ToLowerInvariant();SKEncodedImageFormat format=ext switch{".jpg" or ".jpeg"=>SKEncodedImageFormat.Jpeg,".png"=>SKEncodedImageFormat.Png,".webp"=>SKEncodedImageFormat.Webp,\_=>SKEncodedImageFormat.Png};var skChart=new SKCartesianChart(chartControl){Width=(int)chartControl.ActualWidth,Height=(int)chartControl.ActualHeight};skChart.SaveImage(path,format);MessageBox.Show($"График сохранён в формате {format.ToString().ToLower()} по адресу:\n{path}","Экспорт завершён",MessageBoxButton.OK,MessageBoxImage.Information);}catch(Exception ex){MessageBox.Show(ex.Message,"Ошибка",MessageBoxButton.OK,MessageBoxImage.Error);}finally{InitTheme();}}}public class SalesStatisticsViewModel : INotifyPropertyChanged{#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion private readonly DatabaseContext Db=new();private DbSet<Sale>DbSet;private ObservableCollection<Sale>Sales;private bool \_isGroupedByYear=false;public bool IsGroupedByYear{get=>\_isGroupedByYear;set{if(value!=\_isGroupedByYear){\_isGroupedByYear=value;CreateGraphics();OnPropertyChanged(nameof(IsGroupedByYear));}}}private bool \_isCountGraph=true;public bool IsCountGraph{get=>\_isCountGraph;set{if(value!=\_isCountGraph){\_isCountGraph=value;CreateGraphics();OnPropertyChanged(nameof(IsCountGraph));}}}private record YearMonth(int Year,int Month);private List<IGrouping<YearMonth,Sale>>GroupSales=>[..Sales.GroupBy(e=>new YearMonth(e.Date.Year,e.Date.Month)).OrderBy(g=>g.Key.Year).ThenBy(g=>g.Key.Month)];private List<IGrouping<int,Sale>>GroupSalesByYear=>[..Sales.GroupBy(e=>e.Date.Year).OrderBy(g=>g.Key)];private ISeries[] \_series=[];public ISeries[] Series{get=>\_series;set=>SetProperty(ref \_series,value);}private ICartesianAxis[] \_xAxes=[];public ICartesianAxis[] XAxes{get=>\_xAxes;set=>SetProperty(ref \_xAxes,value);}public ICartesianAxis[] YAxes{get;set;}=[ new Axis{CrosshairLabelsBackground=SKColors.DarkOrange.AsLvcColor(),CrosshairLabelsPaint=new SolidColorPaint(SKColors.DarkRed),CrosshairPaint=new SolidColorPaint(SKColors.DarkOrange,1),CrosshairSnapEnabled=true}];public ICommand RefreshCommand{get;set;}public SalesStatisticsViewModel(){RefreshCommand=new RelayCommand(Refresh);LoadData();if(Sales==null||DbSet==null)throw new ArgumentNullException(nameof(Sales));CreateGraphics();}private void Refresh(object obj){LoadData();CreateGraphics();}private void LoadData(){DbSet=Db.Set<Sale>();DbSet.Include(e=>e.Contract).ThenInclude(c=>c!=null ? c.Product : null).ThenInclude(p=>p!=null ? p.Unit : null).Include(e=>e.Contract).ThenInclude(c=>c!=null ? c.Counterpartie : null).Load();Sales=DbSet.Local.ToObservableCollection();}private ISeries[] CreateSalesSeries<TKey>(IEnumerable<IGrouping<TKey,Sale>>groupedSales){List<ISeries>series=[];List<int>totalSaleCount=[];List<double>totalSaleSumm=[];foreach(var group in groupedSales){int count=0;double summ=0;foreach(var s in group){count+=s.Count;summ+=s.Summ;}totalSaleCount.Add(count);totalSaleSumm.Add(summ);}if(IsCountGraph)series.Add(new LineSeries<int>{Name="Всего продаж",Values=totalSaleCount});else series.Add(new LineSeries<double>{Name="Всего продаж "+Properties.Settings.Default.Currency,Values=totalSaleSumm});return [..series];}private void CreateGraphics(){if(!IsGroupedByYear){Series=CreateSalesSeries(GroupSales);XAxes=[ new Axis{Labeler=value=>value.ToString("N2"),Labels=[..GroupSales.Select(g=>new DateTime(g.Key.Year,g.Key.Month,1).ToString("MMMM\r\nyyyy",new CultureInfo("ru-RU")))]}];}else{Series=CreateSalesSeries(GroupSalesByYear);XAxes=[ new Axis{Labeler=value=>value.ToString("N2"),Labels=[..GroupSalesByYear.Select(g=>g.Key.ToString())]}];}foreach(var axis in XAxes){axis.MinLimit=null;axis.MaxLimit=null;}foreach(var axis in YAxes){axis.MinLimit=null;axis.MaxLimit=null;}}}}<Page x:Class="VacTrack.ViewReport.StockBalanceReport" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" xmlns:local="clr-namespace:VacTrack.ViewReport" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" mc:Ignorable="d" d:DesignHeight="450" d:DesignWidth="800" Title="StockBalanceReport"><Page.DataContext><local:StockBalanceReportViewModel/></Page.DataContext><Page.InputBindings><KeyBinding Key="P" Modifiers="Control" Command="{Binding PrintCommand}"/></Page.InputBindings><Grid><DockPanel VerticalAlignment="Top" Height="40"><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RefreshCommand}" ToolTip="Обновить"><materialDesign:PackIcon Kind="Refresh"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding PrintCommand}" ToolTip="Печать отчёта"><materialDesign:PackIcon Kind="Printer"/></Button><Button Background="{DynamicResource MaterialDesignTextFieldBoxBackground}" Style="{StaticResource MaterialDesignIconButton}" Width="30" Height="30" Margin="10,0,0,0" Command="{Binding RotateDocCommand}" ToolTip="Смена ориентации страницы"><materialDesign:PackIcon Kind="FileRotateRightOutline"/></Button><ListBox Style="{StaticResource MaterialDesignFilterChipPrimaryListBox}" Height="46" Margin="20,0,0,0"><ListBoxItem Content="Группировка" IsSelected="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Итого группы" IsSelected="{Binding IsGroupTotalEnabled}" IsEnabled="{Binding IsGroupingEnabled}"/><ListBoxItem Content="Общие итоги" IsSelected="{Binding AreOverallTotalsEnabled}"/></ListBox><ComboBox ItemsSource="{Binding MaterialLocation}" Style="{StaticResource MaterialDesignFloatingHintComboBox}" DisplayMemberPath="Name" materialDesign:ComboBoxAssist.ShowSelectedItem="True" SelectedItem="{Binding FilterByLocation}" Margin="20,0,0,0" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Фильтр по месту хранения" materialDesign:TextFieldAssist.HasClearButton="True"/><TextBlock DockPanel.Dock="Right" Height="25" Text="{Binding Message}" ToolTip="{Binding Message}" Foreground="{Binding MessageBrush}" Margin="10,0,0,0"/></DockPanel><FlowDocumentReader Document="{Binding Document}" VerticalAlignment="Stretch" HorizontalAlignment="Stretch" ViewingMode="Scroll" Foreground="{DynamicResource MaterialDesignBody}" Margin="0,40,0,0" BorderThickness="5" BorderBrush="#19737171"/></Grid></Page>namespace VacTrack.ViewReport{public partial class StockBalanceReport : Page,ICachedPage{private StockBalanceReportViewModel ThisViewModel=>(StockBalanceReportViewModel)DataContext;public StockBalanceReport()=>InitializeComponent();public void OnNavigatedFromCache()=>ThisViewModel.OpenFromCache();}public class StockBalanceReportViewModel : BaseReportViewModel<Material>{public ObservableCollection<Location>? MaterialLocation{get;set;}#region properties private bool \_IsGroupingEnabled=true;public bool IsGroupingEnabled{get=>\_IsGroupingEnabled;set{SetProperty(ref \_IsGroupingEnabled,value);Refresh(null);}}private bool \_IsGroupTotalEnabled=true;public bool IsGroupTotalEnabled{get=>\_IsGroupTotalEnabled;set{SetProperty(ref \_IsGroupTotalEnabled,value);Refresh(null);}}private bool \_AreOverallTotalsEnabled=true;public bool AreOverallTotalsEnabled{get=>\_AreOverallTotalsEnabled;set{SetProperty(ref \_AreOverallTotalsEnabled,value);Refresh(null);}}private Location? \_FilterByLocation;public Location? FilterByLocation{get=>\_FilterByLocation;set{SetProperty(ref \_FilterByLocation,value);LoadData();}}#endregion protected override void LoadData(){MaterialLocation=new ObservableCollection<Location>([..Db.Locations]);DbSet=Db.Set<Material>();DbSet.Include(e=>e.Unit).Include(e=>e.Location).ThenInclude(e=>e!=null ? e.Employee : null).Load();Items=new ObservableCollection<Material>(DbSet.Local.Where(item=>FilterByLocation==null||item.Location?.Id==FilterByLocation.Id).ToList());}public override FlowDocument CreateReport(){FlowDocument doc=new(){FontFamily=new FontFamily("Times New Roman"),FontSize=12,PagePadding=new Thickness(50)};AddHeader(ref doc);Table table=new(){CellSpacing=3,BorderBrush=Brushes.Gray,BorderThickness=new Thickness(1)};AddTableHeader(ref table);TableRowGroup dataGroup=new();double totalSum=0.0;if(IsGroupingEnabled)CreateGroupedRows(ref dataGroup,ref totalSum);else CreateNoGroupedRows(ref dataGroup,ref totalSum);if(AreOverallTotalsEnabled)dataGroup.Rows.Add(CreateRow(["Итого","","","",$"{totalSum:N2}"]));table.RowGroups.Add(dataGroup);doc.Blocks.Add(table);Employee? responsible=Db.Employees.Include(e=>e.Post).FirstOrDefault(e=>e.Id==Properties.Settings.Default.ResponsibleStorekeeper);doc.Blocks.Add(new Paragraph(new Run(FilterByLocation!=null ? $" \_\_\_\_\_\_\_\_\_\_\_\_\_ { FilterByLocation.Employee?.Fio}" : $"{responsible?.Post?.Name} \_\_\_\_\_\_\_\_\_\_\_\_\_ {responsible?.Fio}")){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5});return doc;}private void CreateGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){CreateGroupedRows(ref dataGroup,item=>item.Location?.Name,item=>item.GetSum,key=>["",$"{key}","","",""],total=>["Итого","","","",$"{total:N2}"],item=>[ $"{item.Name}",String.Empty,$"{item.Count} {item.Unit?.Name}",$"{item.Price:N2}",$"{item.GetSum:N2}" ],ref totalSum,IsGroupTotalEnabled);}private void CreateNoGroupedRows(ref TableRowGroup dataGroup,ref double totalSum){foreach(var item in Items){double summ=item.GetSum;totalSum+=summ;dataGroup.Rows.Add(CreateRow([ $"{item.Name}",$"{item.Location?.Name}",$"{item.Count} {item.Unit?.Name}",$"{item.Price:N2}",$"{summ:N2}" ]));}}private static void AddTableHeader(ref Table table){TableRowGroup headerGroup=new();TableRow headerRow=new();headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Наименование"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Место хранения"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Количество"))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Стоимость,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerRow.Cells.Add(new TableCell(new Paragraph(new Run("Сумма,\n"+Properties.Settings.Default.Currency))){FontWeight=FontWeights.Bold});headerGroup.Rows.Add(headerRow);table.RowGroups.Add(headerGroup);}private void AddHeader(ref FlowDocument doc){Paragraph header=new(new Run("ООО \"ВакТайм\"")){FontSize=16,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(header);Paragraph title=new(new Run("Отчет по остаткам материалов и комплектующих")){FontSize=14,FontWeight=FontWeights.Bold,TextAlignment=TextAlignment.Center};doc.Blocks.Add(title);string filteredText=FilterByLocation!=null ? $"\nПо месту хранения: {FilterByLocation.Name}, Ответственный: {FilterByLocation.Employee?.Fio}" : string.Empty;string headerText=$"Дата формирования: {DateTime.Now:dd.MM.yyyy}"+filteredText;Paragraph headerParagraph=new(new Run(headerText)){FontSize=12,TextAlignment=TextAlignment.Left,LineHeight=1.5};doc.Blocks.Add(headerParagraph);}}}namespace VacTrack.Validators{public class LengthRangeValidationRule : ValidationRule{public int MinLength{get;set;}=0;public int MaxLength{get;set;}=int.MaxValue;public int Length{get;set;}=0;public override ValidationResult Validate(object value,CultureInfo cultureInfo){var input=(value ?? "").ToString();if(Length==0){if(input.Length<MinLength)return new ValidationResult(false,$"Не может быть менее {MinLength} символов.");if(input.Length>MaxLength)return new ValidationResult(false,$"Не может превышать {MaxLength} символов.");}else{if(input.Length!=Length)return new ValidationResult(false,$"Должно быть ровно {Length} символов.");}return ValidationResult.ValidResult;}}}namespace VacTrack.Validators{class LoginCheck : ValidationRule{public override ValidationResult Validate(object value,CultureInfo cultureInfo){var login=value as string ?? string.Empty;if(string.IsNullOrWhiteSpace(login))return new ValidationResult(false,"Логин не может быть пустым.");if(login.Length<4)return new ValidationResult(false,"Логин должен содержать минимум 4 символа.");if(!Regex.IsMatch(login,@"^[а-яА-Яa-zA-Z0-9\_]+$"))return new ValidationResult(false,"Логин должен содержать только буквы, цифры и подчёркивания.");using(var db=new DatabaseContext())if(db.Users.Any(u=>u.Login==login))return new ValidationResult(false,"Такой пользователь уже есть.");return ValidationResult.ValidResult;}}}namespace VacTrack.Validators{public class NotEmptyValidationRule : ValidationRule{public override ValidationResult Validate(object value,CultureInfo cultureInfo){return string.IsNullOrWhiteSpace((value ?? "").ToString())? new ValidationResult(false,"Не может быть пустым."): ValidationResult.ValidResult;}}}namespace VacTrack.Validators{public class PasswordCheck : ValidationRule{public string? FirstPassword{get;set;}public override ValidationResult Validate(object value,CultureInfo cultureInfo){var password=value as string ?? string.Empty;if(FirstPassword!=null&&password!=FirstPassword)return new ValidationResult(false,"Пароли не совпадают.");if(string.IsNullOrWhiteSpace(password))return new ValidationResult(false,"Пароль не может быть пустым.");if(password.Length<6)return new ValidationResult(false,"Пароль должен содержать минимум 6 символов.");if(!Regex.IsMatch(password,@"[A-Z]"))return new ValidationResult(false,"Пароль должен содержать хотя бы одну заглавную букву.");if(!Regex.IsMatch(password,@"[0-9]"))return new ValidationResult(false,"Пароль должен содержать хотя бы одну цифру.");return ValidationResult.ValidResult;}}}namespace VacTrack.Validators{public class PhoneNumberValidationRule : ValidationRule{private readonly string[] \_patterns={@"^\+?[1-9]\d{1,14}$",@"^(?:\+7|8)\d{10}$",@"^\+?\d{1,4}[-\s]?\(?\d{1,3}\)?[-\s]?\d{1,4}[-\s]?\d{1,4}$"};public override ValidationResult Validate(object value,CultureInfo cultureInfo){var input=(value ?? "").ToString();if(string.IsNullOrWhiteSpace(input)){return new ValidationResult(false,"Номер телефона не может быть пустым.");}foreach(var pattern in \_patterns){if(Regex.IsMatch(input,pattern)){return ValidationResult.ValidResult;}}return new ValidationResult(false,"Введите корректный номер телефона.");}}}namespace VacTrack.Validators{public class PositiveNumberValidationRule : ValidationRule{public override ValidationResult Validate(object value,CultureInfo cultureInfo){string? input=(value ?? "").ToString();if(string.IsNullOrWhiteSpace(input))return new ValidationResult(false,"Значение не может быть пустым.");if(double.TryParse(input,NumberStyles.Float,cultureInfo,out double result)){if(result>0)return ValidationResult.ValidResult;else return new ValidationResult(false,"Число должно быть строго положительным.");}return new ValidationResult(false,"Некорректное числовое значение.");}}}namespace VacTrack.Samples{internal class SampleContract{public static FlowDocument GenerateContract(String ConractNo,string contractDate,string term,string price,string address,string counterpartieData,string OrganizationData){FlowDocument doc=new FlowDocument();doc.PagePadding=new Thickness(50);doc.PageWidth=800;Style header1Style=new Style(typeof(Paragraph));header1Style.Setters.Add(new Setter(Paragraph.FontSizeProperty,16.0));header1Style.Setters.Add(new Setter(Paragraph.FontWeightProperty,FontWeights.Bold));header1Style.Setters.Add(new Setter(Paragraph.TextAlignmentProperty,TextAlignment.Center));header1Style.Setters.Add(new Setter(Paragraph.MarginProperty,new Thickness(0,20,0,10)));doc.Resources.Add("header1Style",header1Style);Style header2Style=new Style(typeof(Paragraph));header2Style.Setters.Add(new Setter(Paragraph.FontSizeProperty,14.0));header2Style.Setters.Add(new Setter(Paragraph.FontWeightProperty,FontWeights.Bold));header2Style.Setters.Add(new Setter(Paragraph.MarginProperty,new Thickness(0,15,0,5)));header2Style.Setters.Add(new Setter(Paragraph.TextAlignmentProperty,TextAlignment.Center));doc.Resources.Add("header2Style",header2Style);Style normalStyle=new Style(typeof(Paragraph));normalStyle.Setters.Add(new Setter(Paragraph.FontSizeProperty,12.0));normalStyle.Setters.Add(new Setter(Paragraph.MarginProperty,new Thickness(0,5,0,5)));normalStyle.Setters.Add(new Setter(Paragraph.TextAlignmentProperty,TextAlignment.Justify));doc.Resources.Add("normalStyle",normalStyle);Style boldStyle=new Style(typeof(Paragraph));boldStyle.Setters.Add(new Setter(Paragraph.FontSizeProperty,12.0));boldStyle.Setters.Add(new Setter(Paragraph.FontWeightProperty,FontWeights.Bold));boldStyle.Setters.Add(new Setter(Paragraph.MarginProperty,new Thickness(0,5,0,5)));doc.Resources.Add("boldStyle",boldStyle);Paragraph title=new Paragraph(new Run($"ДОГОВОР ПОСТАВКИ №{ConractNo}"));title.Style=(Style)doc.Resources["header1Style"];doc.Blocks.Add(title);Table headerTable=new Table();headerTable.CellSpacing=0;headerTable.Margin=new Thickness(0,0,0,20);headerTable.Columns.Add(new TableColumn(){Width=new GridLength(1,GridUnitType.Star)});headerTable.Columns.Add(new TableColumn(){Width=new GridLength(1,GridUnitType.Star)});TableRowGroup group=new TableRowGroup();headerTable.RowGroups.Add(group);TableRow row1=new TableRow();group.Rows.Add(row1);TableCell cityCell=new TableCell(new Paragraph(new Run("г. Сморгонь")));cityCell.BorderThickness=new Thickness(0,0,0,1);cityCell.BorderBrush=Brushes.Black;cityCell.Padding=new Thickness(5);cityCell.TextAlignment=TextAlignment.Left;row1.Cells.Add(cityCell);TableCell dateCell=new TableCell(new Paragraph(new Run(contractDate)));dateCell.BorderThickness=new Thickness(0,0,0,1);dateCell.BorderBrush=Brushes.Black;dateCell.Padding=new Thickness(5);dateCell.TextAlignment=TextAlignment.Right;row1.Cells.Add(dateCell);doc.Blocks.Add(headerTable);Paragraph section1Header=new Paragraph(new Run("1. ПРЕДМЕТ ДОГОВОРА"));section1Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section1Header);Paragraph p11=new Paragraph();p11.Style=(Style)doc.Resources["normalStyle"];p11.Inlines.Add(new Run("1.1. Поставщик обязуется поставить Комплектующие к вакуумной установке "));p11.Inlines.Add(new Run("(далее - Товар), а Покупатель - принять и оплатить товар, в соответствии "));p11.Inlines.Add(new Run("со Счёт-протоколом согласования договорной цены (Приложение №1), "));p11.Inlines.Add(new Run("являющегося неотъемлемой частью настоящего договора."));doc.Blocks.Add(p11);Paragraph p12=new Paragraph(new Run("1.2. Покупатель обязуется оплатить и принять Товар."));p12.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p12);Paragraph p13=new Paragraph();p13.Style=(Style)doc.Resources["normalStyle"];p13.Inlines.Add(new Run("1.3. Срок поставки -- "));p13.Inlines.Add(new Bold(new Run(term)));p13.Inlines.Add(new Run("."));doc.Blocks.Add(p13);Paragraph p14=new Paragraph(new Run("1.4. Цель приобретения Товара - для собственного потребления и производства."));p14.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p14);Paragraph section2Header=new Paragraph(new Run("2. СТОИМОСТЬ ТОВАРА И ПОРЯДОК РАСЧЕТОВ"));section2Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section2Header);Paragraph p21=new Paragraph();p21.Style=(Style)doc.Resources["normalStyle"];p21.Inlines.Add(new Run("2.1. Стоимость Товара по данному договору, в соответствии со "));p21.Inlines.Add(new Run("Счет-протоколом (Приложение №1) составляет "));p21.Inlines.Add(new Bold(new Run(price)));p21.Inlines.Add(new Run("."));doc.Blocks.Add(p21);Paragraph p22=new Paragraph(new Run("2.2. Покупатель перечисляет на расчетный счет сумму в размере -- 100% стоимости товара."));p22.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p22);Paragraph p23=new Paragraph(new Run("2.3 Цена поставляемого товара остается фиксированной на протяжении действия договора. В цену товара включается тара, упаковка и транспортировка."));p23.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p23);Paragraph p24=new Paragraph(new Run("2.4. Источник финансирования - собственные средства Покупателя."));p24.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p24);Paragraph p25=new Paragraph(new Run("2.5. Датой оплаты считается дата списания денежных средств Покупателя."));p25.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p25);Paragraph p26=new Paragraph(new Run("2.6. Любые изменения по условиям оплаты согласовываются обеими Сторонами и оформляются дополнительным соглашением."));p26.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p26);Paragraph section3Header=new Paragraph(new Run("3. ПОРЯДОК СДАЧИ И ПРИЕМКИ ТОВАРА"));section3Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section3Header);Paragraph p31=new Paragraph(new Run("3.1. Датой поставки Товара считается дата подписи Покупателем товарно-транспортной накладной о принятии Товара."));p31.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p31);Paragraph p32=new Paragraph(new Run("3.2. От даты поставки Товара Покупателю на него переходит право собственности на Товар, риск его случайной гибели или случайного повреждения."));p32.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p32);Paragraph p33=new Paragraph();p33.Style=(Style)doc.Resources["normalStyle"];p33.Inlines.Add(new Run("3.3. Адрес поставки: "));p33.Inlines.Add(new Bold(new Run(address)));p33.Inlines.Add(new Run(" (склад Покупателя)."));doc.Blocks.Add(p33);Paragraph p34=new Paragraph(new Run("3.4. Допускается досрочная поставка Товара."));p34.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p34);Paragraph p35=new Paragraph(new Run("3.5. Приемка товара по количеству и качеству, проверка работоспособности Товара осуществляются в присутствии представителей Поставщика и Покупателя и подписании технического акта приемки товара."));p35.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p35);Paragraph p36=new Paragraph(new Run("3.6. Приемка Товара по количеству осуществляется на соответствие количества фактически полученного Товара данным, указанным в товарно-транспортной накладной. В случае обнаружения в ходе приемки недостачи Товара Стороны обязаны: приостановить приемку; принять меры по обеспечению сохранности Товара; оформить факт выявленной недостачи актом, подписанным Сторонами."));p36.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p36);Paragraph section4Header=new Paragraph(new Run("4. КАЧЕСТВО ТОВАРА И ГАРАНТИЯ"));section4Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section4Header);Paragraph p41=new Paragraph(new Run("4.1. Качество Товара должно соответствовать техническим требованиям Покупателя."));p41.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p41);Paragraph p42=new Paragraph(new Run("4.2. Упаковка товара должна соответствовать требованиям стандартов, обеспечивающих полную сохранность товара при транспортировке всеми видами транспорта."));p42.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p42);Paragraph p43=new Paragraph(new Run("4.3. Поставщик гарантирует качества товара в течение 12 месяцев с даты приемки товара по товарно-транспортной (товарной) накладной. Срок замены или ремонта товара по гарантийным обязательствам составляет 90 дней с момента передачи неисправного товара по дефектному акту Поставщику, если иное не оговорено сторонами."));p43.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p43);Paragraph p44=new Paragraph(new Run("4.4. На время ремонта поставщик продлевает гарантию на товар."));p44.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p44);Paragraph p45=new Paragraph(new Run("4.5. Поставщик не несет гарантийные обязательства в случае, если условия эксплуатации товара не соответствуют требованиям эксплуатационной документации на товар, если имеются механические повреждения товара вследствие неправильного его хранения или эксплуатации."));p45.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p45);Paragraph section5Header=new Paragraph(new Run("5. ОТВЕТСТВЕННОСТЬ СТОРОН"));section5Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section5Header);Paragraph p51=new Paragraph(new Run("5.1. За неисполнение или ненадлежащее исполнение условий договора Стороны несут ответственность, предусмотренную действующим законодательством Республики Беларусь."));p51.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p51);Paragraph p52=new Paragraph(new Run("5.2. В случае просрочки поставки товара Поставщик уплачивает в пользу Покупателя пеню в размере 0,1% от стоимости не поставленного в срок товара, предусмотренного настоящим договором, за каждый день просрочки выполнения обязательства, но не более 10°/о от стоимости не поставленного товара в срок."));p52.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p52);Paragraph p53=new Paragraph(new Run("5.3. В случае нарушения сроков окончательного расчета за поставленный товар Покупатель уплачивает в пользу Поставщика пеню в размере 0,1% от суммы просроченного платежа, за каждый день просрочки выполнения обязательства, но не более 10% от суммы просроченного платежа."));p53.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p53);Paragraph p54=new Paragraph(new Run("5.4. Уплата неустойки (штрафы, пени) не освобождает стороны от выполнения обязательств по настоящему договору."));p54.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p54);Paragraph p55=new Paragraph(new Run("5.5. Споры, не урегулированные Сторонами в претензионном порядке, передаются на разрешение Экономического суда по месту нахождения ответчика."));p55.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p55);Paragraph p56=new Paragraph(new Run("5.6. Во всем остальном, что не предусмотрено настоящим договором, Стороны руководствуются действующим законодательством Республики Беларусь."));p56.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p56);Paragraph section6Header=new Paragraph(new Run("6. СРОК ДЕЙСТВИЯ ДОГОВОРА"));section6Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section6Header);Paragraph p61=new Paragraph(new Run("6.1. Настоящий договор вступает в силу с момента его подписания Сторонами и действует до полного выполнения Сторонами всех обязательств по настоящему договору."));p61.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p61);Paragraph p62=new Paragraph(new Run("6.2. Документы, переданные по факсимильной связи, имеют юридическую силу."));p62.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p62);Paragraph p63=new Paragraph(new Run("6.3. Все изменения и дополнения к настоящему договору имеют юридическую силу только в том случае, если они совершены в письменной форме и подписаны уполномоченными на то представителями Сторон."));p63.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p63);Paragraph section7Header=new Paragraph(new Run("7. ФОРС-МАЖОР"));section7Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section7Header);Paragraph p71=new Paragraph(new Run("7.1. Стороны освобождаются от ответственности по выполнению обязательств по настоящему договору, если оно явилось следствием непреодолимой силы, а именно: пожар, наводнение, землетрясение, война, боевые действия, блокада, эмбарго на импорт и экспорт. При этом срок выполнения обязательств по настоящему договору отодвигается соразмерно времени, в течение которого действовали эти обстоятельства и их последствия."));p71.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p71);Paragraph p72=new Paragraph(new Run("7.2. Сторона, для которой создалась невозможность выполнения обязательств по договору, обязана немедленно известить другую Сторону о наступлении и прекращении вышеуказанных обстоятельств."));p72.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p72);Paragraph p73=new Paragraph(new Run("7.3. Надлежащим доказательством наличия указанных выше обстоятельств и их продолжительности будет служить свидетельство соответствующих торговых палат."));p73.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p73);Paragraph p74=new Paragraph(new Run("7.4. Если эти обстоятельства и их последствия будут длиться более 4-х месяцев, то каждая из Сторон будет вправе аннулировать договор полностью или частично, и в этом случае ни одна из Сторон не будет иметь права потребовать от другой Стороны возмещения возможных убытков."));p74.Style=(Style)doc.Resources["normalStyle"];doc.Blocks.Add(p74);Paragraph section8Header=new Paragraph(new Run("8. ЮРИДИЧЕСКИЕ АДРЕСА СТОРОН"));section8Header.Style=(Style)doc.Resources["header2Style"];doc.Blocks.Add(section8Header);Table partiesTable=new Table();partiesTable.CellSpacing=0;partiesTable.Margin=new Thickness(0,10,0,20);partiesTable.BorderBrush=Brushes.Transparent;partiesTable.BorderThickness=new Thickness(0);TableRowGroup partiesGroup=new TableRowGroup();partiesTable.RowGroups.Add(partiesGroup);TableRow headerRow=new TableRow();partiesGroup.Rows.Add(headerRow);TableCell buyerHeaderCell=new TableCell(new Paragraph(new Run("Покупатель")));buyerHeaderCell.BorderBrush=Brushes.Transparent;buyerHeaderCell.BorderThickness=new Thickness(0);buyerHeaderCell.Padding=new Thickness(5);buyerHeaderCell.TextAlignment=TextAlignment.Center;headerRow.Cells.Add(buyerHeaderCell);TableCell supplierHeaderCell=new TableCell(new Paragraph(new Run("Поставщик")));supplierHeaderCell.BorderBrush=Brushes.Transparent;supplierHeaderCell.BorderThickness=new Thickness(0);supplierHeaderCell.Padding=new Thickness(5);supplierHeaderCell.TextAlignment=TextAlignment.Center;headerRow.Cells.Add(supplierHeaderCell);TableRow contentRow=new TableRow();partiesGroup.Rows.Add(contentRow);TableCell buyerCell=new TableCell();buyerCell.BorderBrush=Brushes.Transparent;buyerCell.BorderThickness=new Thickness(0);buyerCell.Padding=new Thickness(5);Paragraph buyerPara=new Paragraph(new Run(counterpartieData));buyerPara.Style=(Style)doc.Resources["normalStyle"];buyerCell.Blocks.Add(buyerPara);contentRow.Cells.Add(buyerCell);TableCell supplierCell=new TableCell();supplierCell.BorderBrush=Brushes.Transparent;supplierCell.BorderThickness=new Thickness(0);supplierCell.Padding=new Thickness(5);Paragraph supplierPara=new Paragraph(new Run(OrganizationData));supplierPara.Style=(Style)doc.Resources["normalStyle"];supplierCell.Blocks.Add(supplierPara);;contentRow.Cells.Add(supplierCell);doc.Blocks.Add(partiesTable);return doc;}}}<Window x:Class="VacTrack.DialogWindows.ReceiptsAdd" 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:VacTrack.DialogWindows" xmlns:validators="clr-namespace:VacTrack.Validators" xmlns:converter="clr-namespace:VacTrack.Converter" mc:Ignorable="d" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" Title="Добавление" Height="450" Width="800"><Window.DataContext><local:ReceiptsAddVM/></Window.DataContext><Window.Resources><converter:DockPanelWidthConverter x:Key="DockPanelWidthConverter" Length="300"/></Window.Resources><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><Grid.RowDefinitions><RowDefinition Height="150"/><RowDefinition Height="1\*"/><RowDefinition Height="35"/></Grid.RowDefinitions><FrameworkElement x:Name="DataContextProxy" DataContext="{Binding}"/><StackPanel Margin="15,15,15,15"><StackPanel Orientation="Horizontal" Margin="0,0,0,5"><Label Content="Контрагент:" Width="150" VerticalAlignment="Center"/><ComboBox Width="200" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="От кого" ItemsSource="{Binding ReceiptCounterpartie}" DisplayMemberPath="Name" Style="{StaticResource MaterialDesignFloatingHintComboBox}"><ComboBox.SelectedValue><Binding Path="SelectCounterpartie" UpdateSourceTrigger="PropertyChanged"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></ComboBox.SelectedValue></ComboBox></StackPanel><StackPanel Orientation="Horizontal" Margin="0,0,0,5"><Label Content="Дата:" Width="150" VerticalAlignment="Center"/><DatePicker Width="200" materialDesign:HintAssist.Hint="Дата поступления" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" Foreground="{DynamicResource MaterialDesignBody}"><DatePicker.SelectedDate><Binding Path="SelectDate" UpdateSourceTrigger="PropertyChanged"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></DatePicker.SelectedDate></DatePicker></StackPanel><StackPanel Orientation="Horizontal" Margin="0,0,0,5"><Label Content="Склад назначения:" Width="150" VerticalAlignment="Center"/><ComboBox Width="200" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Склад" ItemsSource="{Binding ReceiptLocation}" DisplayMemberPath="Name" Style="{StaticResource MaterialDesignFloatingHintComboBox}"><ComboBox.SelectedValue><Binding Path="SelectLocation" UpdateSourceTrigger="PropertyChanged"><Binding.ValidationRules><validators:NotEmptyValidationRule/></Binding.ValidationRules></Binding></ComboBox.SelectedValue></ComboBox></StackPanel></StackPanel><DataGrid ItemsSource="{Binding Items}" SelectedItem="{Binding SelectedItem}" AutoGenerateColumns="False" CanUserAddRows="False" HeadersVisibility="All" BorderThickness="3" BorderBrush="#19737171" Grid.Row="1" Margin="5,0,5,0"><DataGrid.Columns><DataGridCheckBoxColumn Binding="{Binding IsSelected, UpdateSourceTrigger=PropertyChanged}" EditingElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridCheckBoxColumnStyle}"><DataGridCheckBoxColumn.Header><Border Background="Transparent"><CheckBox IsChecked="{Binding DataContext.IsAllSelected, Source={x:Reference DataContextProxy}}"/></Border></DataGridCheckBoxColumn.Header><DataGridCheckBoxColumn.HeaderStyle><Style TargetType="{x:Type DataGridColumnHeader}" BasedOn="{StaticResource MaterialDesignDataGridColumnHeader}"><Setter Property="HorizontalContentAlignment" Value="Center"/></Style></DataGridCheckBoxColumn.HeaderStyle></DataGridCheckBoxColumn><materialDesign:DataGridComboBoxColumn Header="Материал" IsEditable="True" ItemsSourceBinding="{Binding RelativeSource={RelativeSource FindAncestor, AncestorType={x:Type DataGrid}}, Path=DataContext.ReceiptMaterial, UpdateSourceTrigger=PropertyChanged}" SelectedValueBinding="{Binding Material}" DisplayMemberPath="Name"></materialDesign:DataGridComboBoxColumn><DataGridTextColumn IsReadOnly="True" Header="Ед.Изм." Binding="{Binding Material.Unit.Name, UpdateSourceTrigger=PropertyChanged}"/><DataGridTextColumn IsReadOnly="True" Header="Цена" Binding="{Binding Material.Price, UpdateSourceTrigger=PropertyChanged}"/><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Количество"><materialDesign:DataGridTextColumn.Binding><Binding Path="Count"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn><materialDesign:DataGridTextColumn EditingElementStyle="{StaticResource MaterialDesignDataGridTextColumnPopupEditingStyle}" ElementStyle="{StaticResource MaterialDesignDataGridTextColumnStyle}" Header="Cумма"><materialDesign:DataGridTextColumn.Binding><Binding Path="Summ" UpdateSourceTrigger="PropertyChanged"><Binding.ValidationRules><validators:PositiveNumberValidationRule/></Binding.ValidationRules></Binding></materialDesign:DataGridTextColumn.Binding></materialDesign:DataGridTextColumn></DataGrid.Columns></DataGrid><DockPanel LastChildFill="False" VerticalAlignment="Bottom" Height="35" x:Name="MainDockPanel" Grid.Row="2" Margin="5,0,5,0"><Button Content="Добавить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding AddCommand}" Width="95" ToolTip="добавление новой записи"/><Button Content="Удалить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" Command="{Binding DeleteCommand}" Width="95" ToolTip="удаление выбраной записи"/><TextBlock HorizontalAlignment="Center" VerticalAlignment="Center" ToolTip="{Binding Message}" Text="{Binding Message}" Foreground="{Binding MessageBrush}" TextTrimming="CharacterEllipsis" Margin="5,0,3,0" MaxWidth="{Binding ActualWidth, ElementName=MainDockPanel, Converter={StaticResource DockPanelWidthConverter}}"/><Button Content="Подвердить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Width="95" ToolTip="Подверждает добаление поступлений" Click="AcceptClick"/><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Margin="0,0,5,0" DockPanel.Dock="Right" Cursor="Hand" Width="95" ToolTip="Подверждает добаление поступлений" Click="CancelClick"/></DockPanel></Grid></Window>namespace VacTrack.DialogWindows{public partial class ReceiptsAdd : Window{private ReceiptsAddVM ThisViewModel=>(ReceiptsAddVM)DataContext;public ObservableCollection<Receipt>? NewItems;public ReceiptsAdd(){InitializeComponent();}private void AcceptClick(object sender,RoutedEventArgs e){if(ThisViewModel.SelectLocation!=null&&ThisViewModel.SelectCounterpartie!=null){ThisViewModel.Message=string.Empty;NewItems=ThisViewModel.Items;DialogResult=NewItems!=null;}else{ThisViewModel.Message="Не все поля заполнены";ThisViewModel.MessageBrush=Brushes.Red;}}private void CancelClick(object sender,RoutedEventArgs e)=>DialogResult=false;}public class ReceiptsAddVM : ViewTables.BaseViewModel<Receipt>{public ObservableCollection<Counterpartie>ReceiptCounterpartie{get;set;}public ObservableCollection<Material>ReceiptMaterial{get;set;}public ObservableCollection<Location>ReceiptLocation{get;set;}private Counterpartie? \_selectCounterpartie;public Counterpartie? SelectCounterpartie{get=>\_selectCounterpartie;set{\_selectCounterpartie=value;foreach(var item in Items)item.Counterpartie=value;OnPropertyChanged(nameof(SelectCounterpartie));}}public Location? SelectLocation{get;set;}public DateTime SelectDate{get;set;}public ReceiptsAddVM(): base(new DatabaseContext()){var MaterDbSet=Db.Set<Material>();MaterDbSet.Include(m=>m.Unit).Load();ReceiptMaterial=MaterDbSet.Local.ToObservableCollection();ReceiptCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);ReceiptLocation=new ObservableCollection<Location>([..Db.Locations]);SelectDate=DateTime.Now;}protected override void LoadData(){DbSet=Db.Set<Receipt>();Items=[];}protected override Receipt CreateNewItem()=>new(){Date=SelectDate,Counterpartie=SelectCounterpartie};protected override bool FilterItem(Receipt item,string? searchText)=>false;}}<Window x:Class="VacTrack.DialogWindows.ReceiptsPrint" 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:VacTrack.DialogWindows" xmlns:validators="clr-namespace:VacTrack.Validators" mc:Ignorable="d" xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes" Style="{StaticResource CustomWindowStyle}" Title="Печать накладной" Height="270" Width="420" ResizeMode="NoResize"><Window.DataContext><local:ReceiptsPrintVM/></Window.DataContext><Grid Background="{DynamicResource MaterialDesign.Brush.Background}"><StackPanel Margin="10,10,10,50"><StackPanel Orientation="Horizontal"><Label Content="Дата:" Width="100" VerticalAlignment="Center"/><DatePicker Width="130" materialDesign:HintAssist.Hint="c" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" Foreground="{DynamicResource MaterialDesignBody}" SelectedDate="{Binding SelectStartDate}" Margin="0,0,10,0" materialDesign:TextFieldAssist.HasClearButton="True"></DatePicker><DatePicker Width="130" materialDesign:HintAssist.Hint="по" Style="{StaticResource MaterialDesignFloatingHintDatePicker}" Foreground="{DynamicResource MaterialDesignBody}" SelectedDate="{Binding SelectEndDate}" materialDesign:TextFieldAssist.HasClearButton="True"></DatePicker></StackPanel><StackPanel Orientation="Horizontal"><Label Content="Склад:" Width="100" VerticalAlignment="Center"/><ComboBox Width="200" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="Место хранения" ItemsSource="{Binding ReceiptLocation}" DisplayMemberPath="Name" Style="{StaticResource MaterialDesignFloatingHintComboBox}" materialDesign:TextFieldAssist.HasClearButton="True" SelectedValue="{Binding SelectLocation, UpdateSourceTrigger=PropertyChanged}"></ComboBox></StackPanel><StackPanel Orientation="Horizontal"><Label Content="Контрагент:" Width="100" VerticalAlignment="Center"/><ComboBox Width="200" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="От кого" ItemsSource="{Binding ReceiptCounterpartie}" DisplayMemberPath="Name" Style="{StaticResource MaterialDesignFloatingHintComboBox}" materialDesign:TextFieldAssist.HasClearButton="True" SelectedValue="{Binding SelectCounterpartie, UpdateSourceTrigger=PropertyChanged}"></ComboBox></StackPanel><StackPanel Orientation="Horizontal" Visibility="Hidden"><Label Content="ТМЦ:" Width="100" VerticalAlignment="Center"/><ComboBox Width="200" Foreground="{DynamicResource MaterialDesignBody}" materialDesign:HintAssist.Hint="ТМЦ" ItemsSource="{Binding ReceiptMaterial}" DisplayMemberPath="Name" Style="{StaticResource MaterialDesignFloatingHintComboBox}" materialDesign:TextFieldAssist.HasClearButton="True" SelectedValue="{Binding SelectMaterial, UpdateSourceTrigger=PropertyChanged}"></ComboBox></StackPanel><TextBlock VerticalAlignment="Center" TextAlignment="Center" Text="{Binding Message, UpdateSourceTrigger=PropertyChanged}" Foreground="{DynamicResource MaterialDesignBody}"/></StackPanel><Button Content="Отмена" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Width="110" ToolTip="отмена" HorizontalAlignment="Left" VerticalAlignment="Bottom" Margin="10,0,0,10" IsCancel="True" Click="Cansel\_Click"/><Button Content="Продолжить" Style="{StaticResource MaterialDesignPaperButton}" Height="25" FontSize="12" Width="110" ToolTip="формирование нкладной" HorizontalAlignment="Right" VerticalAlignment="Bottom" Margin="0,0,10,10" IsDefault="True" Click="Accept\_Click"/></Grid></Window>namespace VacTrack.DialogWindows{public partial class ReceiptsPrint : Window{private ReceiptsPrintVM ThisVM=>(ReceiptsPrintVM)DataContext;public ReceiptsPrintVM? ReceiptsPrintVM{get;private set;}public ObservableCollection<Receipt>? Items{get;set;}public ReceiptsPrint()=>InitializeComponent();private void Cansel\_Click(object sender,RoutedEventArgs e)=>DialogResult=false;private void Accept\_Click(object sender,RoutedEventArgs e){Items=ThisVM.GetItems();if(Items.Count>0){ReceiptsPrintVM=ThisVM;DialogResult=true;}else{ThisVM.Message="Ничего не найдено";}}}public class ReceiptsPrintVM : INotifyPropertyChanged{public ObservableCollection<Counterpartie>ReceiptCounterpartie{get;set;}public ObservableCollection<Location>ReceiptLocation{get;set;}public ObservableCollection<Material>ReceiptMaterial{get;set;}private string? \_message;public string? Message{get=>\_message;set{\_message=value;OnPropertyChanged(nameof(Message));}}#region interface implemented public event PropertyChangedEventHandler? PropertyChanged;protected void SetProperty<Ts>(ref Ts field,Ts value,[CallerMemberName] string? propertyName=null){if(!EqualityComparer<Ts>.Default.Equals(field,value)){field=value;OnPropertyChanged(propertyName);}}protected void OnPropertyChanged([CallerMemberName] string? name=null)=>PropertyChanged?.Invoke(this,new PropertyChangedEventArgs(name));#endregion public DatabaseContext Db;public Counterpartie? SelectCounterpartie{get;set;}public Location? SelectLocation{get;set;}public Material? SelectMaterial{get;set;}public DateTime? SelectStartDate{get;set;}public DateTime? SelectEndDate{get;set;}public ReceiptsPrintVM(){Db=new DatabaseContext();ReceiptCounterpartie=new ObservableCollection<Counterpartie>([..Db.Counterparties]);ReceiptLocation=new ObservableCollection<Location>([..Db.Locations]);ReceiptMaterial=new ObservableCollection<Material>([..Db.Materials]);}public ObservableCollection<Receipt>GetItems(){var DbSet=Db.Set<Receipt>();DbSet.Include(e=>e.Counterpartie).Include(e=>e.Material).ThenInclude(e=>e!=null ? e.Unit : null).Load();return [..DbSet.Local.Where(item=>(SelectCounterpartie==null||item.Counterpartie?.Id==SelectCounterpartie.Id)&&(SelectMaterial==null||item.Material?.Id==SelectMaterial.Id)&&((SelectStartDate==null&&SelectEndDate==null)||(SelectStartDate!=null&&SelectEndDate!=null&&item.Date>=SelectStartDate&&item.Date<=SelectEndDate)||(SelectStartDate!=null&&SelectEndDate==null&&item.Date==SelectStartDate)||(SelectStartDate==null&&SelectEndDate!=null&&item.Date==SelectEndDate))).ToList()];}}}namespace VacTrack.Converter{internal class ContractStatusConverter : IValueConverter{const string created="Создан";const string running="Выполняется";const string completed="Завершен";private static IEnumerable<string>Statuses=>[created,running,completed];public object Convert(object value,Type targetType,object parameter,CultureInfo culture){if(value is ContractStatus type)return type switch{ContractStatus.created=>created,ContractStatus.running=>running,ContractStatus.completed=>completed,\_=>value};if(value is List<ContractStatus>)return Statuses;return value;}public object ConvertBack(object value,Type targetType,object parameter,CultureInfo culture){if(value is string type)return type switch{created=>ContractStatus.created,running=>ContractStatus.running,completed=>ContractStatus.completed,\_=>throw new ArgumentException($"Неизвестный статус: \"{type}\". Проверьте корректность данных.")};throw new ArgumentException($"Ожидалась строка, но получен объект типа {value?.GetType().Name ?? "null"}. Проверьте входные данные.");}}}namespace VacTrack.Converter{class CounterpartieTypeConverter : IValueConverter{const string UrName="Юр. Лицо";const string FizName="Физ. Лицо";private static IEnumerable<string>UrFiz=>[UrName,FizName];public object Convert(object value,Type targetType,object parameter,CultureInfo culture){if(value is CounterpartieType type)return type switch{CounterpartieType.Ur=>UrName,CounterpartieType.Fiz=>FizName,\_=>value};if(value is List<CounterpartieType>)return UrFiz;return value;}public object ConvertBack(object value,Type targetType,object parameter,CultureInfo culture){if(value is string type)return type switch{UrName=>CounterpartieType.Ur,FizName=>CounterpartieType.Fiz,\_=>throw new ArgumentException($"Неизвестный тип контрагента: \"{type}\". Проверьте корректность данных.")};throw new ArgumentException($"Ожидалась строка, но получен объект типа {value?.GetType().Name ?? "null"}. Проверьте входные данные.");}}}namespace VacTrack.Converter{public class DockPanelWidthConverter : IValueConverter{public int Length{get;set;}public DockPanelWidthConverter(){Length=620;}public DockPanelWidthConverter(int length){Length=length;}public object Convert(object value,Type targetType,object parameter,CultureInfo culture){if(value is double totalWidth){double reservedWidth=Length;return Math.Max(0,totalWidth-reservedWidth);}return 0;}public object ConvertBack(object value,Type targetType,object parameter,CultureInfo culture){throw new NotImplementedException();}}}namespace VacTrack.Converter{class InvertBooleanConverter : IValueConverter{public object Convert(object value,Type targetType,object parameter,CultureInfo culture){if(value is bool b)return!b;else return value;}public object ConvertBack(object value,Type targetType,object parameter,CultureInfo culture){if(value is bool b)return!b;else return value;}}}