**Многопользовательское приложение**

**Задание 1**

**1. АРМ работника склада**

**MVVM:**

* Создать ProductModel, WarehouseModel для хранения товаров и склада.
* Разработать WarehouseViewModel, содержащую коллекцию товаров (ObservableCollection<ProductModel>).
* Реализовать команды (ICommand) для добавления/удаления товаров.

**Разделение логики и представления:**

* В WarehouseService реализовать методы загрузки данных.
* В ViewModel использовать WarehouseService, а не обращаться к базе данных напрямую.

**Асинхронное программирование:**

* Реализовать асинхронную загрузку списка товаров с имитацией задержки.
* Кнопка "Загрузить товары" должна отключаться при загрузке.

**Код**

using System;

using System.Collections.Generic;

using System.Collections.ObjectModel;

using System.ComponentModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Input;

namespace Task1

{

internal class WarehouseViewModel : INotifyPropertyChanged

{

private readonly WarehouseService \_warehouseService;

private bool \_isLoading;

public ObservableCollection<ProductModel> Products { get; set; }

public ICommand LoadProductsCommand { get; }

public bool IsLoading

{

get => \_isLoading;

set

{

\_isLoading = value;

OnPropertyChanged(nameof(IsLoading));

}

}

public WarehouseViewModel()

{

\_warehouseService = new WarehouseService();

Products = new ObservableCollection<ProductModel>();

LoadProductsCommand = new RelayCommand(async () => await LoadProductsAsync(), () => !IsLoading);

}

private async Task LoadProductsAsync()

{

IsLoading = true;

var products = await \_warehouseService.LoadProductsAsync();

Products.Clear();

foreach (var product in products)

{

Products.Add(product);

}

IsLoading = false;

}

public event PropertyChangedEventHandler PropertyChanged;

protected void OnPropertyChanged(string propertyName) =>

PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(propertyName));

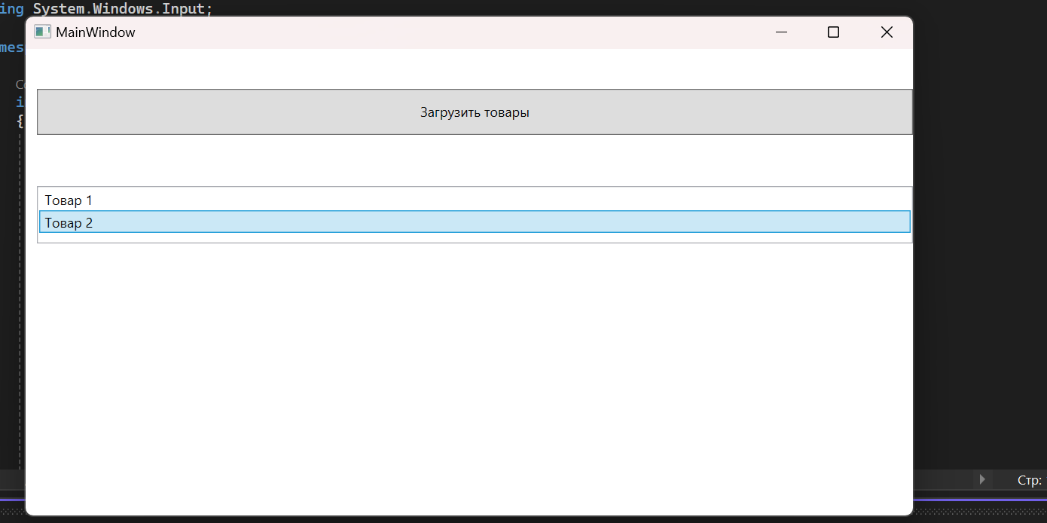
}

}

**Таблица 16 – Входные и выходные данные**

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| **Товар1**  **Товар2** | **Товар1**  **Товар2** |

**Анализ результата**

****