Руководство разработчика

Оглавление

[Руководство разработчика 1](#_Toc122132387)

[**Общие сведения** 2](#_Toc122132388)

[**Запуск Транспортные перевозки** 2](#_Toc122132389)

[**Структура папок** 3](#_Toc122132390)

[**Оформление кода** 3](#_Toc122132391)

[**Модели** 3](#_Toc122132392)

**Общие сведения**

Программа, формализующая работу Перевозок товаров с Aliexpress.

ПО предоставляет:

* Инструменты экспорта;
* Хранение данных;

Транспортные перевозки является побочным продуктом общей деятельности.

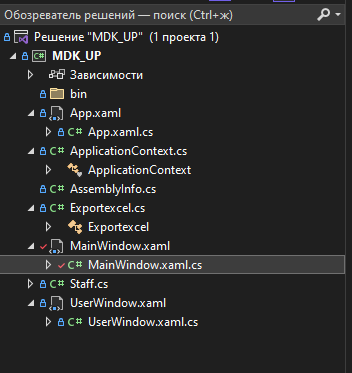
При разработке, один разработчик был одержим следующими идеями:

* Большое количество комментариев к коду
* Получить “Отлично” по УП 02.01
* Много исправлений и ошибок (чтобы на них учится)
* Слушать звук бензопилы (брбррбрбрбрб)

**Запуск Транспортные перевозки**

1. Простой вариант
2. Перейти в репозиторий <https://github.com/akarumai/MDK_UP1.git>
3. Скачать в ZIP формате
4. Разархивируйте в удобное место
5. Скомпилировать проект (Visual Studio) и запустить exe
6. Вариант про
7. Откройте cmd
8. Пропишите git clone <https://github.com/akarumai/MDK_UP1.git> в удобном месте
9. Скомпилировать проект (Visual Studio) и запустить exe

**Структура папок**



**Оформление кода**

Просим придерживаться следующих правил при разработке дополнительных модулей для Перевозок товаров с Aliexpress.

**Модели**

public partial class MainWindow : Window

{

ApplicationContext db = new ApplicationContext();

private string fileName;

public MainWindow()

{

InitializeComponent();

Loaded += MainWindow\_Loaded;

}

// при загрузке окна

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

{

db.Database.EnsureCreated();

db.Staff.Load();

DataContext = db.Staff.Local.ToObservableCollection();

}

// добавление

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

{

UserWindow UserWindow = new UserWindow(new Staff());

if (UserWindow.ShowDialog() == true)

{

Staff Staff = UserWindow.Staff;

db.Staff.Add(Staff);

db.SaveChanges();

}

}

// удаление

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

{

Staff? Staff = usersList.SelectedItem as Staff;

if (Staff is null) return;

db.Staff.Remove(Staff);

db.SaveChanges();

}

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

{

}

private void CreateExcelfile(object sender, RoutedEventArgs e)

{

Exportexcel ExcelExport = new Exportexcel();

ExcelExport.CreateExcelfile(db.Staff.Local.ToObservableCollection());

}

private void CreateJsonfile(object sender, RoutedEventArgs e)

{

var Json = JsonConvert.SerializeObject(DataContext);

string DRpath = "Reports";

if (Directory.Exists(DRpath) == false)

{

Directory.CreateDirectory(DRpath);

}

string exportfile = "Report.json";

DRpath = System.IO.Path.Combine(DRpath, exportfile);

if (File.Exists(DRpath))

File.Delete(DRpath);

FileStream objFileStrm = File.Create(DRpath);

objFileStrm.Close();

File.WriteAllText(DRpath, Json.ToString());

}

}

}