МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Факультет систем управління літальними апаратами

Кафедра 302 «Інформаційні управляючі системи»

**Лабораторна робота № 3**

з дисципліни «Технологія створення програмного продукту»

на тему: «Створення справки для программи»

Виконалs: студентs 2 курсу групи 327 ст

напряму підготовки (спеціальності)

6.050101 Комп’ютерні науки

Крейзе М.В, Лисяк О.Е., Шеин В.А.

Прийняв: доц. каф 302 Момот М.О\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна шкала: \_\_\_\_\_\_\_\_\_\_

Кількість балів: \_\_\_\_\_

Оцінка: ECTS \_\_\_\_\_

Харків – 2018

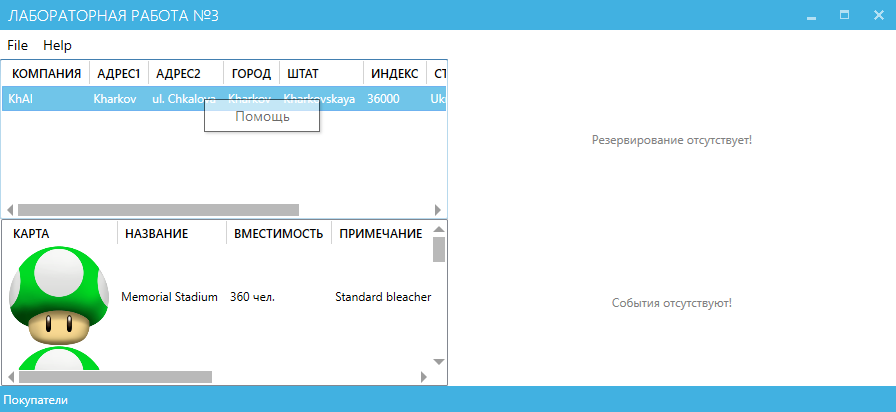
**Лабораторная работа № 3**

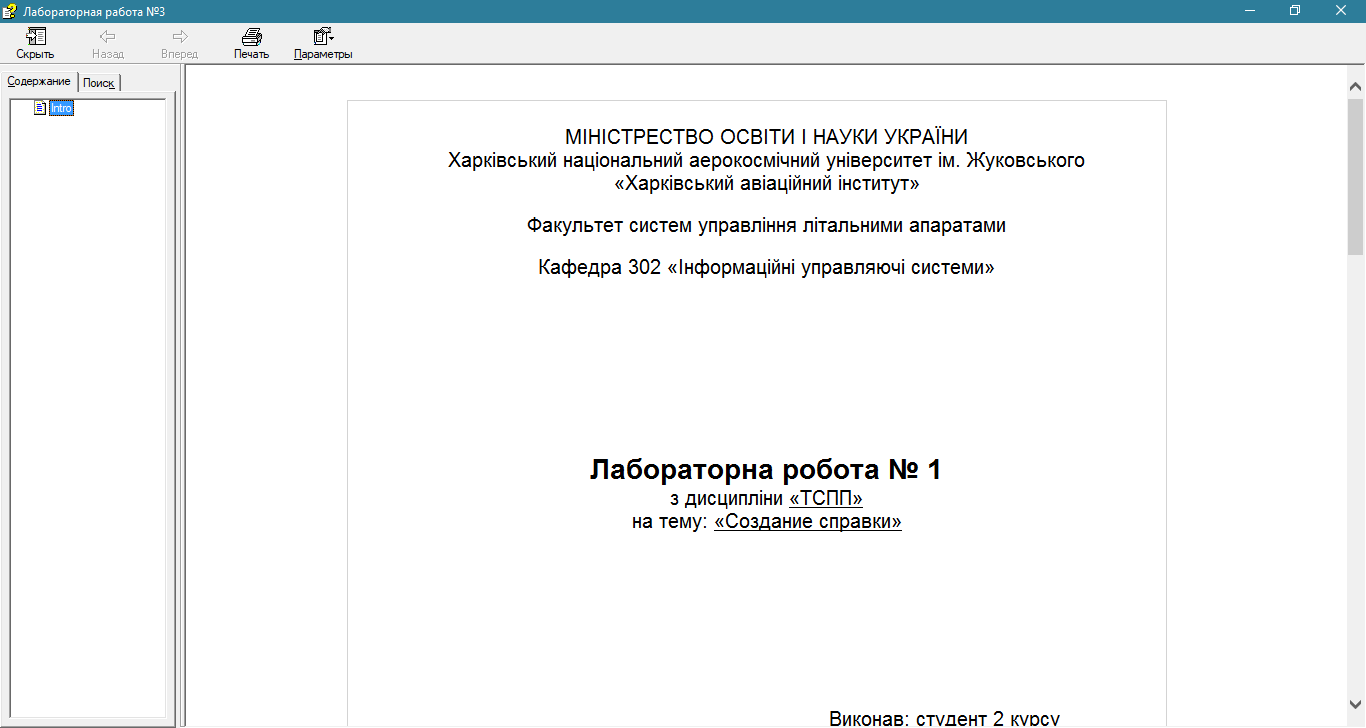
Тема: Создание справки к программе

**Дополнительное задание:**

1. Главная форма содержит меню, в котором есть обязательные пункты File (Exit) Help(About –с информацией о создателях, Help context) – для связи с последующей лабораторной работой.
2. Приложение содержит несколько окон для отображения данных, причем некоторые из них модальные.
3. Формы отображения таблиц содержат управляющие элементы DBNavigator, DBGrid и др. из закладки компонентов Data Controls.
4. Заголовки полей в DBGrid – по-русски.
5. Помеченные \* таблицы имеют графическое поле – отображать содержимое на форме.

**ХОД РАБОТЫ**





**Листинг программы:**

using Lab1.Model;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using MahApps.Metro.Controls;

using MahApps.Metro.Controls.Dialogs;

namespace Lab1

{

/// <summary>

///    MainWindow.xaml

/// </summary>

public partial class MainWindow : MetroWindow, INotifyPropertyChanged

{

private EventContext context = new EventContext();

private List<Venue> venues;

public List<Venue> Venues { get { return venues; } set { venues = value; RaisePropertyChanged("Venues"); } }

private List<Event> events;

public List<Event> Events { get { return events; } set { events = value; RaisePropertyChanged("Events"); } }

private List<Reservation> reservations;

public List<Reservation> Reservations { get { return reservations; } set { reservations = value; RaisePropertyChanged("Reservations"); } }

private List<Customer> customers;

public List<Customer> Customers { get { return customers; } set { customers = value; RaisePropertyChanged("Customers"); } }

public MainWindow()

{

InitializeComponent();

venues = context.Venues.ToList();

customers = context.Customers.ToList();

DataContext = this;

//dataGrid.ItemsSource = c.Venues.ToList();

//dataGrid\_Copy.ItemsSource = c.Events.ToList();

//dataGrid\_Copy2.ItemsSource = c.Reservations.ToList();

//dataGrid\_Copy3.ItemsSource = c.Customers.ToList();

}

#region INotifyPropertyChanged

public event PropertyChangedEventHandler PropertyChanged;

public void RaisePropertyChanged(string propertyName)

{

PropertyChangedEventHandler handler = PropertyChanged;

if (handler != null)

{

PropertyChanged(this, new PropertyChangedEventArgs(propertyName));

}

}

#endregion

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

{

Application.Current.Shutdown();

}

private async void aboutButton\_Click(object sender, RoutedEventArgs e)

{

await this.ShowMessageAsync(" ", "  1\n:\n 327   ", MessageDialogStyle.Affirmative);

}

private void listBox\_MouseDoubleClick(object sender, MouseButtonEventArgs e)

{

try

{

int cID = (listBox\_Copy2.SelectedItem as Customer).CustomerID;

Reservations = context.Reservations.Where(x => x.CustomerID == cID).ToList();

}

catch { }

}

private void listBox\_MouseDoubleClick\_1(object sender, MouseButtonEventArgs e)

{

try

{

int vID = (listBox.SelectedItem as Venue).VenueID;

Events = context.Events.Where(x => x.VenueID == vID).ToList();

}

catch { }

}

private void listBox\_Copy\_MouseDoubleClick(object sender, MouseButtonEventArgs e)

{

try

{

int eID = (listBox\_Copy.SelectedItem as Event).EventID;

Reservations = context.Reservations.Where(x => x.EventID == eID).ToList();

}

catch { }

}

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

{

System.Windows.Forms.Help.ShowHelp(null, @"D:\help.chm");

}

public void HelpOpenByID(int id)

{

System.Windows.Forms.Help.ShowHelp(null, @"D:\help.chm", System.Windows.Forms.HelpNavigator.TopicId, id.ToString());

}

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

{

HelpOpenByID(4);

}

private void MenuItem\_Click\_1(object sender, RoutedEventArgs e)

{

HelpOpenByID(5);

}

private void MenuItem\_Click\_2(object sender, RoutedEventArgs e)

{

HelpOpenByID(6);

}

private void MenuItem\_Click\_3(object sender, RoutedEventArgs e)

{

HelpOpenByID(7);

}

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

{

KeyDown += new KeyEventHandler(MainWindow\_KeyDown);

}

void MainWindow\_KeyDown(object sender, KeyEventArgs e)

{

if (e.Key == Key.F1)

{

UIElement elementWithFocus = Keyboard.FocusedElement as UIElement;

switch (elementWithFocus.ToString())

{

case "System.Windows.Controls.ListViewItem: Lab1.Model.Customer":

MenuItem\_Click(null, null);

break;

case "System.Windows.Controls.ListViewItem: Lab1.Model.Venue":

MenuItem\_Click\_1(null, null);

break;

case "System.Windows.Controls.ListViewItem: Lab1.Model.Event":

MenuItem\_Click\_2(null, null);

break;

case "System.Windows.Controls.ListViewItem: Lab1.Model.Reservation":

MenuItem\_Click\_3(null, null);

break;

default:

HelpContextMenu\_Click(null, null);

break;

}

}

}

private void listBox\_Copy2\_MouseEnter(object sender, MouseEventArgs e)

{

status.Text = "";

}

private void listBox\_MouseEnter(object sender, MouseEventArgs e)

{

status.Text = " ";

}

private void listBox\_Copy\_MouseEnter(object sender, MouseEventArgs e)

{

status.Text = "";

}

private void listBox\_Copy1\_MouseEnter(object sender, MouseEventArgs e)

{

status.Text = "";

}

private void listBox\_Copy2\_MouseLeave(object sender, MouseEventArgs e)

{

status.Text = "";

}

}

}

**ВЫВОД:**

Во время выполнения лабораторной работы мы научились создавать chm(help) файлы.