

## Лабораторная работа №35

### Создание проекта с использованием компонентов стандартных диалогов

#### 1 Цель работы

1.1 Изучить процесс создания и применения стандартных диалоговых окон в приложениях WPF.

#### 2 Литература

2.1 <https://metanit.com/sharp/wpf/20.3.php> – Диалоговые окна

#### 3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

#### 4 Основное оборудование

4.1 Персональный компьютер.

#### 5 Задание

5.1 Использование диалогового окна для открытия файла

5.1.1 Создать WPF-приложение. Добавить в приложение:

- поле ввода с возможностью многострочного ввода,
- кнопку «Открыть».

5.1.2 При нажатии на кнопку «Открыть» предоставить пользователю возможность выбора текстового файла (допустимые расширения: txt, cs, html, css, js, sql).

5.1.3 Если ShowDialog() у диалогового окна вернул true, то:

- загружать текст выбранного файла в поле ввода,
- изменять заголовок окна на название выбранного файла.

5.2 Использование диалогового окна для сохранения файла

5.2.1 Добавить в приложение из п.5.1 кнопку «Сохранить».

При нажатии на кнопку «Сохранить» предоставить пользователю возможность указания имени текстового файла (допустимые расширения: txt, cs, html, css, js, sql).

Для сохранения использовать SaveFileDialog. У диалогового окна настроить фильтрацию на допустимые расширения.

5.2.2 Если ShowDialog() у диалогового окна вернул true, то:

- записывать данные из поля ввода в указанный файл,
- сообщить пользователю о сохранении.

5.3 Настройка MessageBox

5.3.1 Добавить в приложение из п.5.1 обработчик события окна Closing.

5.3.2 В обработчике события Closing создать MessageBox, в котором настроить текст сообщения «Вы хотите закрыть приложение?», заголовок «Подтверждение», иконку с вопросительным знаком и кнопки Да/Нет.

5.3.3 Если пользователь не нажал «Да», то для отмены закрытия написать:  
e.Cancel = true;

5.4 Использование диалогового окна для открытия файлов

5.4.1 Создать WPF-приложение. Добавить в приложение:

- кнопку «Открыть»
- селектор ListView с DataTemplate:  
<Image Source="{Binding .}" Height="100"/>

У селектора изменить панель на Wrap.

5.4.2 При нажатии на кнопку «Открыть» предоставить пользователю возможность выбора изображений. Для открытия использовать OpenFileDialog.

У диалогового окна настроить:

- фильтрацию на расширения png, jpg (jpeg), bmp,
- разрешение на выбор нескольких файлов (Multiselect).

5.4.3 Если ShowDialog() у диалогового окна вернул true, то изменить источник данных селектора на полученный массив названий файлов.

## 5.5 Использование диалогового окна для печати файла

5.5.1 Создать WPF-приложение. Добавить в приложение:

- кнопку «Печать»,
- элемент-контейнер (нужен, т.к. PrintDialog учитывает отступы относительно контейнера, в котором находится печатаемый элемент),
- разместить в элемент-контейнере другой контейнер (например, Canvas) с именем.

В Canvas разместить произвольные элементы.

5.5.2 При нажатии на кнопку «Печать» открыть PrintDialog.

Если ShowDialog() у диалогового окна вернул true, то с помощью окна печати сохранить содержимое именованного контейнера в формате pdf или xps:

окноПечати.PrintVisual(печатаемыйОбъект, "описание"); // здесь объект – Canvas

## 6Порядок выполнения работы

6.1 Выполнить все задания из п.5 в одном решении LabWork35. Каждый проект – приложение WPF.

6.2 Ответить на контрольные вопросы.

## 7Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

## 8Контрольные вопросы

8.1 Что такое диалоговое окно?

8.2 Для чего используется OpenFileDialog?

8.3 Для чего используется SaveFileDialog?

8.4 Что такое MessageBox и какие настройки можно ему задать?

8.5 Для чего используется PrintDialog?