Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.03 «Программирование в компьютерных системах»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

*(наименование профессионального модуля)*

Выполнил:

обучающийся учебной группы № 420

Якунькина София

*(И.О. Фамилия)*

**Москва**

**20­­­­22**

Специальность 09.02.03 «Программирование в компьютерных системах»

*(наименование специальности)*

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

*(наименование профессионального модуля)*

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 7**

**Тема: «События C#».**

Студент: Якунькина София

Группа: ПКС-420

Преподаватель: Сибирев И. В.

**Цель работы**: разработать калькулятор на языке С#. В проекте использовать прием создания одного обработчика для нескольких событий.



Рисунок 1. Пример работы калькулятора

Изображение выглядит как текст, электроника, калькулятор

Автоматически созданное описание

Рисунок 2. Пример работы при выполнении недопустимого действия

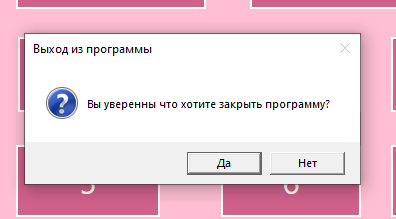


Рисунок 3. Выход из программы

**Код программы:**

<Window x:Class="MyCalculatorv1.MainWindow" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" Title="WPF Calculator" Height="663" Width="708" ResizeMode="NoResize" WindowStartupLocation="CenterScreen" Topmost="True">

<Grid Background="#FFFFBED3" Height="653" VerticalAlignment="Top">

<Grid.RowDefinitions>

<RowDefinition/>

<RowDefinition Height="0\*"/>

<RowDefinition Height="0\*"/>

</Grid.RowDefinitions>

<Button Background="#FFD0618A" Content="7" HorizontalAlignment="Left" Margin="22,224,0,0" VerticalAlignment="Top" Width="132" Height="75" FontSize="32" BorderThickness="2,2,2,1" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<TextBox Height="74" TextWrapping="Wrap" VerticalAlignment="Top" Margin="0,20,0,0" HorizontalAlignment="Center" Width="664" FontSize="40" Background="#FFD0618A" BorderThickness="2" Name="tb" Text="" Foreground="White"/>

<Button Background="#FFD0618A" Content="8" HorizontalAlignment="Left" Margin="212,224,0,0" VerticalAlignment="Top" Width="140" Height="75" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="9" HorizontalAlignment="Left" Margin="415,224,0,0" VerticalAlignment="Top" Width="140" Height="75" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="/" HorizontalAlignment="Left" Margin="585,224,0,0" VerticalAlignment="Top" Width="83" Height="75" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="4" HorizontalAlignment="Left" Margin="22,332,0,0" VerticalAlignment="Top" Width="132" Height="72" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="5" HorizontalAlignment="Left" Margin="210,332,0,0" VerticalAlignment="Top" Width="144" Height="72" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="6" HorizontalAlignment="Left" Margin="415,332,0,0" VerticalAlignment="Top" Width="140" Height="72" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" RenderTransformOrigin="0.524,0.607" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="\*" HorizontalAlignment="Left" Margin="585,332,0,0" VerticalAlignment="Top" Width="83" Height="72" FontSize="40" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="1" HorizontalAlignment="Left" Margin="22,436,0,0" VerticalAlignment="Top" Width="132" Height="74" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="2" HorizontalAlignment="Left" Margin="210,436,0,0" VerticalAlignment="Top" Width="144" Height="74" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="3" HorizontalAlignment="Left" Margin="415,436,0,0" VerticalAlignment="Top" Width="140" Height="74" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="-" HorizontalAlignment="Left" Margin="585,436,0,0" VerticalAlignment="Top" Width="83" Height="74" FontSize="40" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="0" HorizontalAlignment="Left" Margin="22,546,0,0" VerticalAlignment="Top" Width="132" Height="71" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="=" HorizontalAlignment="Left" Margin="210,546,0,0" VerticalAlignment="Top" Width="345" Height="71" FontSize="32" BorderThickness="2" Click="Result\_click" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="+" HorizontalAlignment="Left" Margin="585,546,0,0" VerticalAlignment="Top" Width="83" Height="71" FontSize="32" BorderThickness="2" Click="Button\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="Off" HorizontalAlignment="Left" Margin="444,112,0,0" VerticalAlignment="Top" Width="224" Height="84" FontSize="32" BorderThickness="2" Click="Off\_Click\_1" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Background="#FFD0618A" Content="Del" HorizontalAlignment="Left" Margin="22,112,0,0" VerticalAlignment="Top" Width="154" Height="84" FontSize="32" BorderThickness="2" Click="Del\_Click" Foreground="White">

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

<Button Content="R" HorizontalAlignment="Left" Margin="210,112,0,0" VerticalAlignment="Top" Width="190" Height="84" FontSize="32" BorderThickness="2" Click="R\_Click" Foreground="White" Background="#FFD0618A" >

<Button.BorderBrush>

<SolidColorBrush Color="{DynamicResource {x:Static SystemColors.WindowColorKey}}"/>

</Button.BorderBrush>

</Button>

</Grid>

</Window>

using System;

using System.Windows;

using System.Windows.Controls;

namespace MyCalculatorv1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

Button b = (Button)sender;

tb.Text += b.Content.ToString();

}

private void Result\_click(object sender, RoutedEventArgs e)

{

try

{

result();

}

catch (Exception exc)

{

tb.Text = "Error!";

}

}

private void result()

{

String op;

int iOp = 0;

if (tb.Text.Contains("+"))

{

iOp = tb.Text.IndexOf("+");

}

else if (tb.Text.Contains("-"))

{

iOp = tb.Text.IndexOf("-");

}

else if (tb.Text.Contains("\*"))

{

iOp = tb.Text.IndexOf("\*");

}

else if (tb.Text.Contains("/"))

{

iOp = tb.Text.IndexOf("/");

}

else

{

//error

}

op = tb.Text.Substring(iOp, 1);

double op1 = Convert.ToDouble(tb.Text.Substring(0, iOp));

double op2 = Convert.ToDouble(tb.Text.Substring(iOp + 1, tb.Text.Length - iOp - 1));

if (op == "+")

{

tb.Text += "=" + (op1 + op2);

}

else if (op == "-")

{

tb.Text += "=" + (op1 - op2);

}

else if (op == "\*")

{

tb.Text += "=" + (op1 \* op2);

}

else

{

tb.Text += "=" + (op1 / op2);

}

}

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

{

if (MessageBox.Show("Вы уверенны что хотите закрыть программу?", "Выход из программы", MessageBoxButton.YesNo, MessageBoxImage.Question) == MessageBoxResult.Yes)

{

Close();

}

}

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

{

tb.Text = "";

}

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

{

if (tb.Text.Length > 0)

{

tb.Text = tb.Text.Substring(0, tb.Text.Length - 1);

}

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 8**

**Тема: Проверка орфографии**

Студент: Якунькина София

Группа: ПКС-420

Преподаватель: Сибирев И. В.

**Цель работы**: разработать программу, которая предлагает пользователю ввести какие-либо слова, предложения в текстовое поле и после нажатия соответствующей кнопки проверить орфографию введенного текста. Для непосредственной проверки орфографии воспользуемся функцией Checkspelling объектной библиотеки MS Word.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. Пример вида программы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. Пример работы программы

**Код программы:**

<Window x:Class="WpfApp2.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:WpfApp2"

mc:Ignorable="d"

Title="SpellCheck" MinHeight="460" MaxHeight="460" MinWidth="600" MaxWidth="600">

<Grid Height="auto" Width="auto">

<Grid.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="#FF7A407B" Offset="1"/>

<GradientStop Color="#FF41A1B3" Offset="0"/>

</LinearGradientBrush>

</Grid.Background>

<TextBox Name ="TextBox1" TextWrapping="Wrap" Margin="70,142,70,142" FontSize="16" FontFamily="Comic Sans MS" FontStyle="Normal" FontWeight="Normal" Cursor="IBeam"/>

<Button Name = "SpellCheck" Click= "SpellCheck\_Click" Content="Проверка орфографии" HorizontalAlignment="Center" Margin="0,333,0,0" VerticalAlignment="Top" Height="56" Width="308" FontSize="22" FontFamily="Comic Sans MS" FontWeight="Bold" Background="#FFC0E9FF" BorderBrush="#FF100C0B"/>

<TextBlock HorizontalAlignment="Left" Height="35" Margin="70,87,0,0" VerticalAlignment="Top" Width="266" FontFamily="Comic Sans MS" FontWeight="Bold" FontSize="24" Foreground="Black"><Run Language="ru-ru" Text="Введите текст"/></TextBlock>

</Grid>

</Window>

using System;

using System.Collections.Generic;

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;

namespace WpfApp2

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

TextBox1.Focus();

}

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

{

if (TextBox1.Text.Trim() == "")

{

MessageBox.Show("Вы ввели пустую строку!");

TextBox1.Focus();

}

else

{

TextBox1.SpellCheck.IsEnabled = true;

TextBox1.Focus();

}

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 9**

**Тема: Создание таблицы в MS Word**

Студент: Якунькина София

Группа: ПКС-420

Преподаватель: Сибирев И. В.

**Цель работы**: необходимо разработать программу, которая при нажатии на кнопку создает таблицу в MS Word.

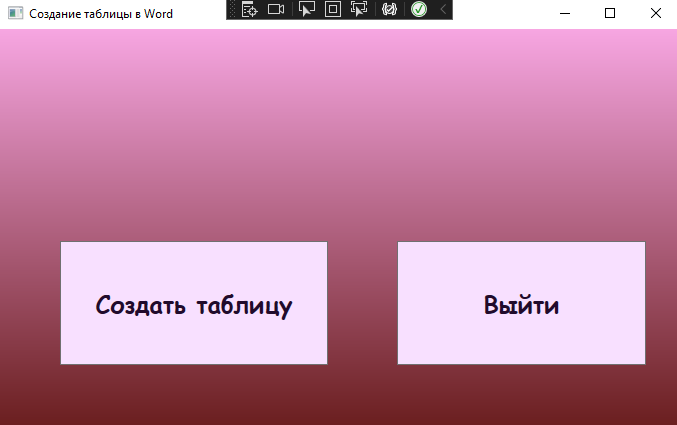


Рисунок 1. Пример вида программы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. Окно сохранения

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 2. Пример сохранённой таблицы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3. Окно выхода из программы

**Код программы:**

<Window x:Class="winTable.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:winTable"

mc:Ignorable="d"

Title="Создание таблицы в Word" MinHeight="440" MaxHeight="440" MinWidth="695" MaxWidth="695">

<Grid>

<Grid.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="#FFF7A6E2"/>

<GradientStop Color="#FF691D1D" Offset="1"/>

</LinearGradientBrush>

</Grid.Background>

<Button Name ="TableButton" Click="TableCreate" Content="Создать таблицу" HorizontalAlignment="Left" Height="124" Margin="60,212,0,0" VerticalAlignment="Top" Width="268" FontSize="24" FontFamily="Comic Sans MS" FontWeight="Bold" Background="#FFF8E0FF" Foreground="#FF210B2C" Cursor="Hand"/>

<Button Name="CloseButton" Click="closeButton\_Click" Content="Выйти" HorizontalAlignment="Left" Height="124" Margin="397,212,0,0" VerticalAlignment="Top" Width="249" FontSize="24" FontFamily="Comic Sans MS" FontWeight="Bold" Background="#FFF8E0FF" Foreground="#FF210B2C" Cursor="Hand"/>

</Grid>

</Window>

using System;

using System.Collections.Generic;

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 Word = Microsoft.Office.Interop.Word;

using Microsoft.Win32;

namespace winTable

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void TableCreate(object sender, RoutedEventArgs e)

{

var application = new Word.Application();

Word.Document document = application.Documents.Add();

Word.Paragraph titleParagraph = document.Paragraphs.Add();

Word.Range titleRange = titleParagraph.Range;

titleRange.Text = "Таблица номеров телефонов";

titleParagraph.set\_Style("Заголовок");

titleRange.InsertParagraphAfter();

Word.Paragraph tableParagraph = document.Paragraphs.Add();

Word.Range tableRange = tableParagraph.Range;

Word.Table smthTable = document.Tables.Add(tableRange, 5, 2);

smthTable.Borders.InsideLineStyle = smthTable.Borders.OutsideLineStyle = Word.WdLineStyle.wdLineStyleSingle;

smthTable.Range.Cells.VerticalAlignment = Word.WdCellVerticalAlignment.wdCellAlignVerticalCenter;

Word.Range cellRange;

cellRange = smthTable.Cell(1, 1).Range;

cellRange.Text = "ID";

cellRange = smthTable.Cell(1, 2).Range;

cellRange.Text = "Номер телефона";

smthTable.Rows[1].Range.Bold = 1;

smthTable.Rows[1].Range.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

for (int i = 0; i < 4; i++)

{

Random rnd = new Random();

int phoneNum = rnd.Next(100000000, 999999999);

cellRange = smthTable.Cell(i + 2, 1).Range;

cellRange.Text = (i+1).ToString();

cellRange = smthTable.Cell(i + 2, 2).Range;

cellRange.Text = "89" + phoneNum.ToString();

}

SaveFileDialog saveFileDialog = new SaveFileDialog();

saveFileDialog.Filter = "Text file (\*.docx)|\*.docx|Pdf file (\*.pdf)|\*.pdf";

if (saveFileDialog.ShowDialog() == true)

document.SaveAs2(saveFileDialog.FileName);

application.Visible = true;

}

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

{

if (MessageBox.Show("Вы уверенны что хотите закрыть программу?", "Выход из программы" , MessageBoxButton.YesNo, MessageBoxImage.Question) == MessageBoxResult.Yes)

{

Close();

}

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 10**

**Тема: Работа с функциями Excel**

Студент: Якунькина София

Группа: ПКС-420

Преподаватель: Сибирев И. В.

**Цель работы**: необходимо создать собственный проект с использованием функций MS Excel.

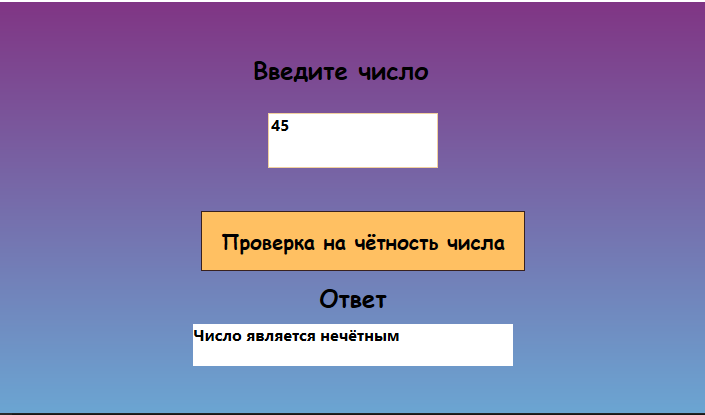


Рисунок 1. Пример работы программы

**Код программы:**

<Window x:Class="ExcelForm.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:ExcelForm"

mc:Ignorable="d"

Title="MainWindow" MaxHeight="450" MinHeight="450" MinWidth="722" MaxWidth="722" Background="LightSkyBlue">

<Grid>

<Grid.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="#FF803584"/>

<GradientStop Color="#FF6BA5D2" Offset="1"/>

</LinearGradientBrush>

</Grid.Background>

<TextBox Name="TextBox" Background="White" HorizontalAlignment="Center" TextWrapping="Wrap" VerticalAlignment="Top" Width="170" Height="55" BorderBrush="#FFF1D09E" SelectionBrush="#FFF1D09E" FontSize="16" FontWeight="Bold" Margin="0,111,0,0"/>

<Button Content="Проверка на чётность числа" HorizontalAlignment="Left" VerticalAlignment="Top" Click="Button\_Click" Height="60" Width="324" FontWeight="Bold" FontSize="20" Background="#FFFFC062" BorderBrush="#FF38211F" Margin="201,209,0,0" FontFamily="Comic Sans MS" Cursor="Hand"/>

<TextBlock Name="TextBlock" Background="White" HorizontalAlignment="Center" Margin="0,322,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Height="42" Width="320" FontSize="16" FontWeight="Bold"/>

<Label Content="Ответ" HorizontalAlignment="Center" Margin="0,274,0,0" VerticalAlignment="Top" FontSize="24" FontWeight="Bold" FontFamily="Comic Sans MS"/>

<TextBlock HorizontalAlignment="Center" Margin="0,51,0,0" TextWrapping="Wrap" Text="Введите число" VerticalAlignment="Top" FontWeight="Bold" FontSize="24" FontFamily="Comic Sans MS" Height="40" Width="200"/>

</Grid>

</Window>

using System;

using System.Collections.Generic;

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;

namespace ExcelForm

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

TextBox.Focus();

}

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

{

if (TextBox.Text.Trim() == "")

{

MessageBox.Show("Вы ввели пустую строку!");

TextBox.Focus();

}

else if (Double.TryParse(TextBox.Text, out double i) == false)

{

MessageBox.Show("Неправильный формат числа!");

TextBox.Focus();

}

else

{

var XL = new Microsoft.Office.Interop.Excel.Application();

var num = XL.WorksheetFunction.IsOdd(Convert.ToDouble(TextBox.Text));

if (num == false)

{

TextBlock.Text = "Число является чётным";

}

else

{

TextBlock.Text = "Число является нечётным";

}

XL.Quit();

}

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 11**

**Тема: Реализация работы с файлами в различных кодировках**

Студент: Якунькина София

Группа: ПКС-420

Преподаватель: Сибирев И. В.

**Цель работы**: написать программу на языке C#, которая реализует работу с файлами в различных кодировках.

Программа должна обладать следующим функционалом:

* Чтение информации из файла в кодировке Unicode
* Чтение информации из файла в кодировке Win1251
* Чтение информации из файла с разрешением RTF
* Чтение бинарных файлов
* Сохранение информации в файл с кодировкой Unicode
* Сохранение информации в файл с кодировкай Win1251
* Сохранение информации в файл с разрешением RTF
* Сохранение информации в бинарный файл
* Печать текстового документа.

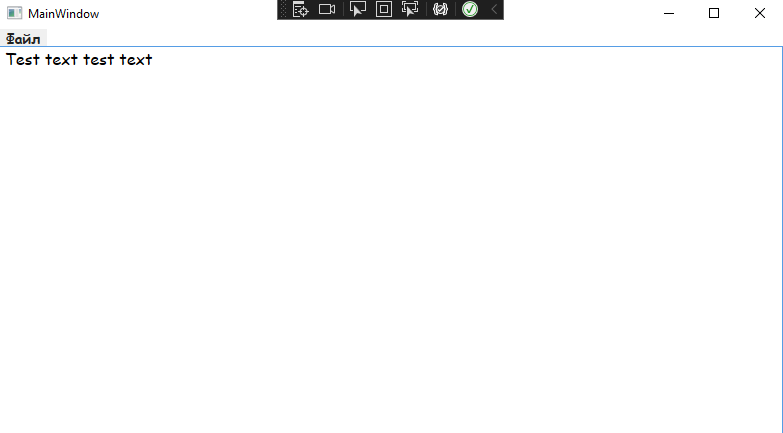


Рисунок 1. Пример программы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. Пример работы программы

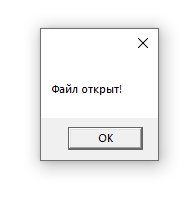
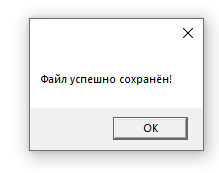


Рисунок 3–4. Сохранение и открытие файлов

**Код программы:**

<Window x:Class="Text\_editor.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:Text\_editor"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Background="White">

<Grid Name="grid1">

<Grid.RowDefinitions>

<RowDefinition Height="177\*"/>

<RowDefinition Height="40\*"/>

</Grid.RowDefinitions>

<Menu Margin="0,-1,0,0" VerticalAlignment="Top" HorizontalAlignment="Left">

<MenuItem Header="Файл" FontWeight="Bold" Height="18" FontFamily="Comic Sans MS" FontSize="14">

<MenuItem Header="Открыть">

<MenuItem Header="Открыть в формате Unicode" Click="Open\_as\_Unicode"></MenuItem>

<MenuItem Header="Открыть в формате Win1251" Click="Open\_as\_Win1251"></MenuItem>

<MenuItem Header="Открыть в формате RTF" Click="Open\_as\_RTF" ></MenuItem>

<MenuItem Header="Открыть бинарный файл" Click="Open\_as\_Binary"></MenuItem>

</MenuItem>

<MenuItem Header="Сохранить как">

<MenuItem Header="в формате RTF" Click="Save\_as\_RTF"></MenuItem>

<MenuItem Header="в кодировке Unicode" Click="Save\_as\_Unicode"></MenuItem>

<MenuItem Header="в кодировке Win1251" Click="Save\_as\_Win1251"></MenuItem>

<MenuItem Header="в бинарный файл" Click="Save\_as\_BinaryFormat"></MenuItem>

</MenuItem>

<MenuItem Header="Распечатать" Click="PrintWindow"></MenuItem>

</MenuItem>

</Menu>

<RichTextBox Name="rtxtbx" Background="White" Width="Auto" Height="Auto" Margin="0,17,0,0" Grid.RowSpan="2" SelectionBrush="#FF3D87B5" FontSize="16" FontFamily="Comic Sans MS">

<FlowDocument>

<Paragraph>

<Run/>

</Paragraph>

</FlowDocument>

</RichTextBox>

</Grid>

</Window>

using Microsoft.Win32;

using System;

using System.IO;

using System.Collections.Generic;

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 Microsoft.SqlServer.Server;

using System.Runtime.Serialization.Formatters.Binary;

namespace Text\_editor

{

public partial class MainWindow : Window

{

Microsoft.Win32.SaveFileDialog dlg = new Microsoft.Win32.SaveFileDialog();

Microsoft.Win32.OpenFileDialog odlg = new Microsoft.Win32.OpenFileDialog();

public MainWindow()

{

InitializeComponent();

dlg.FileName = "Документ";

odlg.FileName = "Документ";

}

private void Save\_as\_RTF(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".rtf";

dlg.Filter = "Текстовый документ (.rtf)|\*.rtf";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText);

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_Unicode(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".txt";

dlg.Filter = "Текстовый документ (.txt)|\*.txt";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText, Encoding.Unicode);

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_Win1251(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".txt";

dlg.Filter = "Текстовый документ (.txt)|\*.txt";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText, Encoding.GetEncoding("windows-1251"));

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_BinaryFormat(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".bin";

dlg.Filter = "Текстовый документ (.bin)|\*.bin";

BinaryFormatter bf = new BinaryFormatter();

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

using (BinaryWriter writer = new BinaryWriter(File.Open(filename, FileMode.OpenOrCreate)))

{

writer.Write(richText);

MessageBox.Show("Файл успешно сохранён!(на рабочий стол)");

}

}

private void Open\_as\_RTF(object sender, RoutedEventArgs e)

{

rtxtbx.Document.Blocks.Clear();

odlg.DefaultExt = ".rtf";

odlg.Filter = "Текстовый документ (.rtf)|\*.rtf";

bool? dialogResult = odlg.ShowDialog();

if (dialogResult.HasValue && dialogResult.Value)

{

TextRange range = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd);

System.IO.FileStream file = new System.IO.FileStream(odlg.FileName, System.IO.FileMode.Open, System.IO.FileAccess.Read);

range.Load(file, System.Windows.DataFormats.Rtf);

file.Dispose();

MessageBox.Show("Файл открыт!");

}

}

private void Open\_as\_Unicode(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename);

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void Open\_as\_Win1251(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename, System.Text.Encoding.GetEncoding("windows-1251"));

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void Open\_as\_Binary(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename);

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void PrintWindow(object sender, RoutedEventArgs e)

{

PrintDialog p = new PrintDialog();

if (p.ShowDialog() == true)

{

p.PrintVisual(grid1, "Печать");

}

}

}

}