**Количество не отрицательное**

<Window x:Class="WpfApp1.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:WpfApp1"

mc:Ignorable="d"

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

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

</Grid.RowDefinitions>

<Label Grid.Row="0" Content="Количество:"/>

<TextBox Grid.Row="0" Margin="80,0,0,0" x:Name="QuantityTextBox" Text="{Binding Quantity, UpdateSourceTrigger=PropertyChanged}"/>

<Button Grid.Row="1" Content="Проверить" Click="CheckButton\_Click" Margin="0,10,0,0"/>

<Label Grid.Row="2" x:Name="ResultLabel"/>

</Grid>

</Window>

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;

namespace WpfApp1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window, INotifyPropertyChanged

{

private int \_quantity;

public int Quantity

{

get { return \_quantity; }

set

{

if (value >= 0)

{

\_quantity = value;

OnPropertyChanged(nameof(Quantity));

}

}

}

public event PropertyChangedEventHandler PropertyChanged;

protected void OnPropertyChanged(string propertyName)

{

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

}

public MainWindow()

{

InitializeComponent();

DataContext = this;

}

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

{

if (Quantity >= 0)

{

ResultLabel.Content = "Количество корректно!";

}

else

{

ResultLabel.Content = "Количество не может быть отрицательным!";

}

}

}

}

**Минимальное кол-во символов**

<Window x:Class="WpfApp1.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:WpfApp1"

mc:Ignorable="d"

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

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

</Grid.RowDefinitions>

<Label Content="Город:"/>

<TextBox x:Name="textBox"

Grid.Row="1"

Margin="5"

PreviewTextInput="TextBox\_PreviewTextInput"/>

<Button x:Name="checkButton"

Grid.Row="2"

Margin="5"

Content="Проверить"

Click="CheckButton\_Click"/>

</Grid>

</Window>

using System;

using System.Text.RegularExpressions;

using System.Windows;

using System.Windows.Input;

namespace WpfApp1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void TextBox\_PreviewTextInput(object sender, TextCompositionEventArgs e)

{

// Разрешаем ввод только букв и пробелов

Regex regex = new Regex("[^a-zA-Zа-яА-Я\\s]+");

e.Handled = regex.IsMatch(e.Text);

}

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

{

// Валидация длины введенного текста

if (textBox.Text.Length < 2 || textBox.Text.Length > 25)

{

MessageBox.Show("Некорректный город!");

textBox.Focus();

}

else

{

MessageBox.Show("Город: " + textBox.Text);

}

}

}

}

**Не должно отрицат, а также четырехразрядное**

<Window x:Class="WpfApp1.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:WpfApp1"

mc:Ignorable="d"

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

<Grid>

<StackPanel Margin="10">

<Label Content="Год издания:"/>

<TextBox x:Name="yearTextBox" Margin="0,5" MaxLength="4" TextChanged="yearTextBox\_TextChanged"/>

<Button Content="Проверить" Margin="0,10" Click="checkButton\_Click"/>

<Label x:Name="resultLabel" Margin="0,10"/>

</StackPanel>

</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 WpfApp1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

int year;

bool isValid = int.TryParse(yearTextBox.Text, out year) && year >= 0 && year < 10000;

resultLabel.Content = isValid ? "Год издания корректен" : "Год издания некорректен";

resultLabel.Foreground = isValid ? Brushes.Green : Brushes.Red;

}

private void yearTextBox\_TextChanged(object sender, TextChangedEventArgs e)

{

int year;

bool isValid = int.TryParse(yearTextBox.Text, out year) && year >= 0 && year < 10000;

yearTextBox.Background = isValid ? Brushes.White : Brushes.LightPink;

}

}

}

**Почт индекс 6 симвл**

<Window x:Class="Проверка.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:Проверка"

mc:Ignorable="d"

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

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

</Grid.RowDefinitions>

<Label Grid.Row="0" Content="Почтовый индекс:" Margin="5"/>

<TextBox Grid.Row="1" Name="zipCodeTextBox" Margin="5"

PreviewTextInput="ZipCodeTextBox\_PreviewTextInput"

MaxLength="6" HorizontalContentAlignment="Center"

Text="{Binding ZipCode, UpdateSourceTrigger=PropertyChanged}"/>

<TextBlock Grid.Row="2" Name="validationMessageTextBlock" Margin="5"

Text="{Binding ValidationMessage}" Foreground="Red"/>

</Grid>

</Window>

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Linq;

using System.Runtime.CompilerServices;

using System.Text;

using System.Text.RegularExpressions;

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 Проверка

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

private string \_zipCode;

private string \_validationMessage;

public string ZipCode

{

get => \_zipCode;

set

{

\_zipCode = value;

OnPropertyChanged();

ValidateZipCode();

}

}

public string ValidationMessage

{

get => \_validationMessage;

set

{

\_validationMessage = value;

OnPropertyChanged();

}

}

public MainWindow()

{

InitializeComponent();

DataContext = this;

}

private void ZipCodeTextBox\_PreviewTextInput(object sender, System.Windows.Input.TextCompositionEventArgs e)

{

e.Handled = !IsZipCodeInputValid(e.Text);

}

private bool IsZipCodeInputValid(string input)

{

return Regex.IsMatch(input, "^[0-9]\*$") && (zipCodeTextBox.Text + input).Length <= 6;

}

private void ValidateZipCode()

{

ValidationMessage = string.IsNullOrEmpty(ZipCode) || ZipCode.Length == 6

? null

: "Почтовый индекс должен содержать 6 цифр.";

}

public event PropertyChangedEventHandler PropertyChanged;

private void OnPropertyChanged([CallerMemberName] string propertyName = null)

{

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

}

}

}

**Проверка телефона ххххх**

<Window x:Class="проверка\_e\_mail.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:проверка\_e\_mail"

mc:Ignorable="d"

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

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<Label Grid.Row="0" Content="Номер телефона:" Margin="10"/>

<TextBox Grid.Row="0" Margin="10" x:Name="PhoneTextBox"

PreviewTextInput="PhoneTextBox\_PreviewTextInput"/>

<Button Grid.Row="1" Margin="10" Content="Проверить" Click="CheckButton\_Click"/>

<Label Grid.Row="2" Margin="10" x:Name="ResultLabel"/>

</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 проверка\_e\_mail

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void PhoneTextBox\_PreviewTextInput(object sender, TextCompositionEventArgs e)

{

// Разрешаем ввод цифр, знака плюса и тире

e.Handled = !char.IsDigit(e.Text[0]) && e.Text[0] != '+' && e.Text[0] != '-';

}

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

{

string phone = PhoneTextBox.Text;

if (IsValidPhone(phone))

{

ResultLabel.Content = "Номер телефона введен правильно.";

}

else

{

ResultLabel.Content = "Номер телефона введен неправильно.";

}

}

private bool IsValidPhone(string phone)

{

// Проверяем, соответствует ли введенный номер телефона маске "+X-XXX-XXX-XX-XX"

return System.Text.RegularExpressions.Regex.IsMatch(phone, @"\+[\d]-[\d]{3}-[\d]{3}-[\d]{2}-[\d]{2}");

}

}

}

**Электронная почта**

<Window x:Class="WpfApp1.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:WpfApp1"

mc:Ignorable="d"

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

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="Auto"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<Label Grid.Row="0" Grid.Column="0" Content="Email:"/>

<TextBox Grid.Row="0" Grid.Column="1" x:Name="emailTextBox"/>

<Label Grid.Row="1" Grid.Column="0"/>

<TextBlock Grid.Row="1" Grid.Column="1" x:Name="validationText"/>

<Button Grid.Row="2" Grid.Column="1" Content="Check" Click="CheckButton\_Click"/>

</Grid>

</Window>

using System.Text.RegularExpressions;

using System.Windows;

namespace WpfApp1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

string email = emailTextBox.Text;

if (Regex.IsMatch(email, "^[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+.[A-Za-z]{2,}$"))

{

validationText.Text = "Правмильный email!";

}

else

{

validationText.Text = "Не правильный email!";

}

}

}

}