

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное**

**учреждение Московской области**

**«Люберецкий техникум имени Героя Советского Союза,**

**лётчика-космонавта Ю.А. Гагарина»**

**КУРСОВАЯ РАБОТА**

**по МДК.01.01 «Разработка программных модулей»**

**по МДК.01.03 «Разработка мобильных приложений»**

специальность

09.02.07 «Информационные системы и программирование»

**Тема:**

Создание нескольких приложений на языке C#

Группа 185

Выполнил/а/ студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Алдошкин Владислав Романович

(подпись) (ФИО полностью)

Руководитель Жирнова Юлия Витальевна

(подпись) (ФИО полностью)

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Люберцы-2021 год

Содержание

[Введение 3](#_Toc95399769)

[Постановка задачи 4](#_Toc95399770)

[Изучение программных средств 4](#_Toc95399771)

[Постановка задачи 5](#_Toc95399772)

[Создание 1 проекта 6](#_Toc95399773)

[Создание счетчика 6](#_Toc95399774)

[Создание генератора случайных чисел 7](#_Toc95399775)

[Создание блокнота 7](#_Toc95399776)

[Создаем генератор паролей 8](#_Toc95399777)

[Создаем конвертер 8](#_Toc95399778)

[Form1.cs 11](#_Toc95399779)

[Генератор пароля и блокнот.cs 12](#_Toc95399780)

[Конвертер.cs 19](#_Toc95399781)

[рандомайзер.cs 29](#_Toc95399782)

[счетчик.cs 33](#_Toc95399783)

[Выводы 36](#_Toc95399784)

[Список Литературы 37](#_Toc95399785)

# Введение

Современный мир теперь нельзя представить без информационных систем. Они встречаются постоянно в нашей жизни. Информационные системы очень помогают в жизни и значительно упрощают ее.

Каждая организация стремиться минимизировать затраты времени, материальных, трудовых ресурсов в ходе своей деятельности и упростить процесс обработки информации. Эти задачи можно решить с использованием информационных систем.

Целью курсовой работы является создание нескольких приложений на языке C#.

**1.Теоретическая часть**

# Постановка задачи

Для изучения новых свойств языка C# необходимо понимать его возможности, для этого и создаются программы следующие программы:

1. Счетчик
2. генератор случайных чисел
3. блокнот
4. генератор паролей
5. конвертер.

# Изучение программных средств

Microsoft Visual Studio — линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов.

**2. Практическая часть**

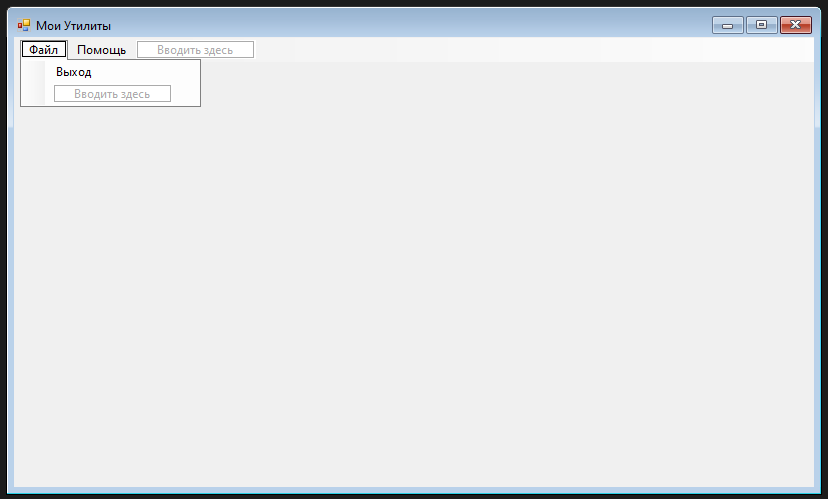
# Постановка задачи

Для изучения новых свойств языка C# необходимо понимать его возможности, для этого и создаются программы следующие программы:

1. Счетчик
2. генератор случайных чисел
3. блокнот
4. генератор паролей
5. конвертер.

# Создание 1 проекта

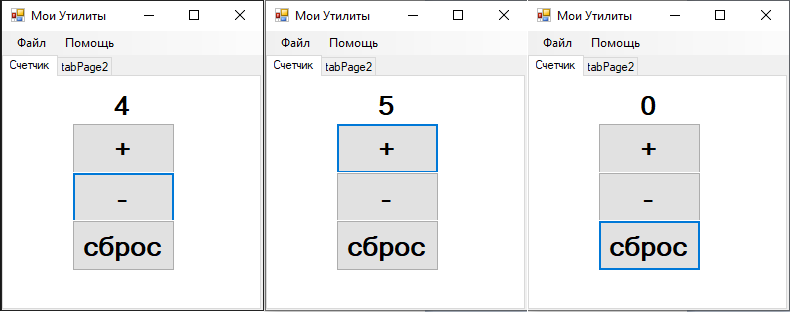
Создание 1 проекта под названием: "MyUtilities". Меняем название формы и текст формы на MainForm и Мои утилиты. Добавляем на форму элемент под названием MenuStrip и создаем пункты и подпункты. Добавляем повидения на пункты ("выход" и о "программе") (Рисунок 1).



(Рисунок 1)

# Создание счетчика

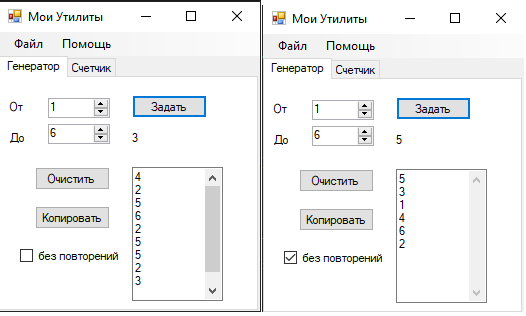
Добавляем в "MyUtilities" элемент под названием TabControl (В свойствах находим Dock и выставляем в значении Fill), указываем названия вкладок и добавляем 3 кнопки + label, меняем их названия. Создание обработчиков для кнопок (Рисунок 2).



(Рисунок 2)

# Создание генератора случайных чисел

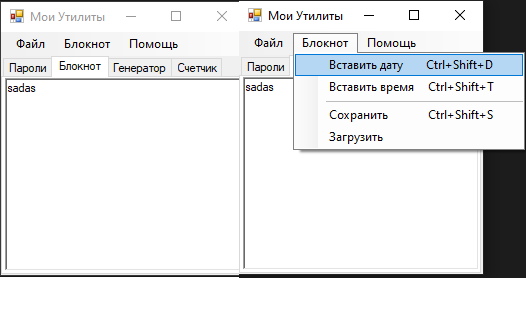
Меняем название окна на генератор, добавляем 6 элементов (Button, 3 label, 2 NumericUpDown). В NumericUpDown меняем начальные значения на 1 и 6. Создаем обработчики для кнопок. Добавляем элемент TextBox и прописываем ему функцию запоминания. Добавляем еще одну кнопку, называем ее очистить и прописываем метод очищения. Добавляем кнопку для копирования чисел в буфер обмена. добавляем исключение повторяющихся чисел. Добавляем переключатель для отключения и включения повторений чисел. Прописываем замену числа при отключенном повторении чисел (Рисунок 3).



(Рисунок 3)

# Создание блокнота

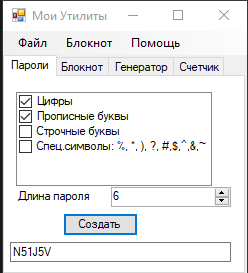
Добавляем страницу и называем ее "Блокнот". Добавляем эллемент "RichTextBox" Выставляем свойство Dock в Fill. Добавляем автоматическое определение даты и времени (Создаем пункт меню и его подпункты). Добавляем обработчики для указания даты и времени в блокноте. Добавляем Hotkey. Добавляем конпки сохранения и загрузки и обработчики событий для них + добавляем Hotkey. Создаем обработчик для формы (открытие последнего сохраненного файла) (Рисунок 4).



(Рисунок 4)

# Создаем генератор паролей

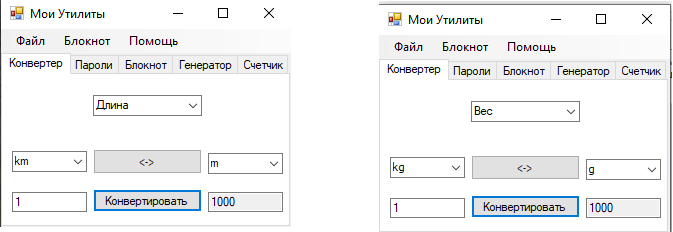
Добавляем страницу и называем ее "Пароли". Добавляем CheckedListBox и заполняем его, добавляем NumericUpDown, label, button, TextBox и форматируем их под себя. Создаем обработчик для рандомной записи символов и массив со спец. символами. Создаем автоматическое копирование в буфер обмена. Добавляем выбор по умолчанию (Рисунок 5).



(Рисунок 5)

# Создаем конвертер

Добавляем страницу и называем ее "Конвертер". Добавляем 2 ComboBox, Button, 2 TextBox и форматируем все элементы под себя, заполняем элементы. в TextBoxTo выставляем значение ReadOnly на True. Создаем словарь (Dictionary) и заполняем его. Создаем обработчик кнопки. Добвляем кнопку смены конвертора и создаем обработчик. Добавляем разные методы конвертирования (вес и тд.) для этого добавляем ComboBox и добавляем обработку по методу SelectedIndexChanged (Рисунок 6).



(Рисунок 6)

# Form1.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MyUtil

{

public partial class MainForm : Form

{

public MainForm()

{

InitializeComponent();

}

private void tsmiExit\_Click(object sender, EventArgs e)

{

this.Close(); //присвоение функции выход

}

private void tsmiAbout\_Click(object sender, EventArgs e)

{

MessageBox.Show("Программа содержит ряд небольших программ," +

"\n которые могут пригодиться в жизни.", "О программе");

//после запятой - название окна; \n - перенос на другую строчку

}

}

}

# Генератор пароля и блокнот.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MyUtil

{

public partial class MainForm : Form

{

int count = 0;

Random rnd;//обработчик рандом

char[] spec\_chars = new char[] { '%', '\*', ')', '?', '#', '$', '^', '&', '~' };//массив спец. символов

public MainForm()

{

InitializeComponent();

rnd = new Random();

}

private void tsmiExit\_Click(object sender, EventArgs e)

{

this.Close(); //присвоение функции выход

}

private void tsmiAbout\_Click(object sender, EventArgs e)

{

MessageBox.Show("Программа содержит ряд небольших программ," +

"\n которые могут пригодиться в жизни.", "О программе");

//после запятой - название окна; \n - перенос на другую строчку

}

private void btnPlus\_Click(object sender, EventArgs e)

{

count++;//складывает в счетчике

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnMinus\_Click(object sender, EventArgs e)

{

count--;//вычитает из счетчика

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnReset\_Click(object sender, EventArgs e)

{

count = 0;//обнуляет счетчик

lblCount.Text = Convert.ToString(count);//преобразование в стринг, но другим способом

}

private void button1\_Click(object sender, EventArgs e)

{

int n;

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);//Создание случайного числа

lblRandom.Text = n.ToString();//Вывод случайного числа в label

if (cbRandom.Checked)//Метод для CheckBox отключение и включение повторения чисел

{

int i = 0;

while (tbRandom.Text.IndexOf(n.ToString()) != -1)//исключение повторяющихся чисел

{

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);

i++;//увеличение счетчика на 1

if (i > 1000) break;//прерывание цикла при попытках больше 1000

}

if (i <= 1000) tbRandom.AppendText(n + " \n");// сохранение числа в случае i <= 1000

}

else

tbRandom.AppendText(n + " \n");//Сохранение чисел в TextBox

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

private void numericUpDown1\_ValueChanged(object sender, EventArgs e)

{

}

private void numericUpDown2\_ValueChanged(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void btnRandomClear\_Click(object sender, EventArgs e)

{

tbRandom.Clear();//очистка TextBox

}

private void btnCopy\_Click(object sender, EventArgs e)

{

Clipboard.SetText(tbRandom.Text);//Копирование в буфер обмена

}

private void tsmiInsertDate\_Click(object sender, EventArgs e)

{

RTBNotepad.AppendText(DateTime.Now.ToShortDateString() + "\n");//Автоматическое определение даты

}

private void tsmiInsertTime\_Click(object sender, EventArgs e)

{

RTBNotepad.AppendText(DateTime.Now.ToShortTimeString() + "\n");// Автоматическое определение времени

}

private void tsmiSave\_Click(object sender, EventArgs e)

{

try

{

RTBNotepad.SaveFile("notepad.rtf");//Сохранение файла

}

catch {

MessageBox.Show("Ошибка при сохранении");//вывод ошибки

}

}

void LoadNotepad()

{

try

{

RTBNotepad.LoadFile("notepad.rtf");//Загрузка файла в блокнот

}

catch

{

MessageBox.Show("Ошибка при Загрузке");//вывод ошибки

}

}

private void tsmiLoad\_Click(object sender, EventArgs e)

{

LoadNotepad();

}

private void MainForm\_Load(object sender, EventArgs e)

{

LoadNotepad();

CLBPassword.SetItemChecked(0, true);//выбор по умолчанию в генерации пароля

CLBPassword.SetItemChecked(1, true);

}

private void BtnCreatePass\_Click(object sender, EventArgs e)

{

if (CLBPassword.CheckedItems.Count == 0) return;//отмена генерации пароля при 0

string password = "";//пустой пароль

for (int i = 0; i < NUDPassLength.Value; i++)//цикл проверки значений

{

int n = rnd.Next(0, CLBPassword.CheckedItems.Count);//случайны выбор элементов из массива

string s = CLBPassword.CheckedItems[n].ToString();//запись выбранного элемента в переменную S

switch (s)

{

case "Цифры":

password += rnd.Next(10).ToString();//запись случайных цифр (от 0 до 9)

break;

case "Прописные буквы":

password += Convert.ToChar(rnd.Next(65, 88));//преобразование чисел в символы

break;

case "Строчные буквы":

password += Convert.ToChar(rnd.Next(97, 122));//преобразование чисел в символы

break;

default:

password += spec\_chars[rnd.Next(spec\_chars.Length)];//запись случайных спец. символов

break;

}

TBPass.Text = password;//присвоение TextBox функции password

Clipboard.SetText(password);//копирование пароля в буфер обмена

}

}

}

}

# Конвертер.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MyUtil

{

public partial class MainForm : Form

{

int count = 0;

Random rnd;//обработчик рандом

char[] spec\_chars = new char[] { '%', '\*', ')', '?', '#', '$', '^', '&', '~' };//массив спец. символов

Dictionary<string, double> metrica;//структура данных, позволяющая получать доступ к элементу по ключу

public MainForm()

{

InitializeComponent();

rnd = new Random();

metrica = new Dictionary<string, double>();//создаем объект metrica

metrica.Add("mm", 1);//заполняем metrica

metrica.Add("cm", 10);

metrica.Add("dm", 100);

metrica.Add("m", 1000);

metrica.Add("km", 1000000);

metrica.Add("mile", 1609344);

}

private void tsmiExit\_Click(object sender, EventArgs e)

{

this.Close(); //присвоение функции выход

}

private void tsmiAbout\_Click(object sender, EventArgs e)

{

MessageBox.Show("Программа содержит ряд небольших программ," +

"\n которые могут пригодиться в жизни.", "О программе");

//после запятой - название окна; \n - перенос на другую строчку

}

private void btnPlus\_Click(object sender, EventArgs e)

{

count++;//складывает в счетчике

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnMinus\_Click(object sender, EventArgs e)

{

count--;//вычитает из счетчика

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnReset\_Click(object sender, EventArgs e)

{

count = 0;//обнуляет счетчик

lblCount.Text = Convert.ToString(count);//преобразование в стринг, но другим способом

}

private void button1\_Click(object sender, EventArgs e)

{

int n;

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);//Создание случайного числа

lblRandom.Text = n.ToString();//Вывод случайного числа в label

if (cbRandom.Checked)//Метод для CheckBox отключение и включение повторения чисел

{

int i = 0;

while (tbRandom.Text.IndexOf(n.ToString()) != -1)//исключение повторяющихся чисел

{

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);

i++;//увеличение счетчика на 1

if (i > 1000) break;//прерывание цикла при попытках больше 1000

}

if (i <= 1000) tbRandom.AppendText(n + " \n");// сохранение числа в случае i <= 1000

}

else

tbRandom.AppendText(n + " \n");//Сохранение чисел в TextBox

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

private void numericUpDown1\_ValueChanged(object sender, EventArgs e)

{

}

private void numericUpDown2\_ValueChanged(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void btnRandomClear\_Click(object sender, EventArgs e)

{

tbRandom.Clear();//очистка TextBox

}

private void btnCopy\_Click(object sender, EventArgs e)

{

Clipboard.SetText(tbRandom.Text);//Копирование в буфер обмена

}

private void tsmiInsertDate\_Click(object sender, EventArgs e)

{

RTBNotepad.AppendText(DateTime.Now.ToShortDateString() + "\n");//Автоматическое определение даты

}

private void tsmiInsertTime\_Click(object sender, EventArgs e)

{

RTBNotepad.AppendText(DateTime.Now.ToShortTimeString() + "\n");// Автоматическое определение времени

}

private void tsmiSave\_Click(object sender, EventArgs e)

{

try

{

RTBNotepad.SaveFile("notepad.rtf");//Сохранение файла

}

catch {

MessageBox.Show("Ошибка при сохранении");//вывод ошибки

}

}

void LoadNotepad()

{

try

{

RTBNotepad.LoadFile("notepad.rtf");//Загрузка файла в блокнот

}

catch

{

MessageBox.Show("Ошибка при Загрузке");//вывод ошибки

}

}

private void tsmiLoad\_Click(object sender, EventArgs e)

{

LoadNotepad();

}

private void MainForm\_Load(object sender, EventArgs e)

{

LoadNotepad();

CLBPassword.SetItemChecked(0, true);//выбор по умолчанию в генерации пароля

CLBPassword.SetItemChecked(1, true);

}

private void BtnCreatePass\_Click(object sender, EventArgs e)

{

if (CLBPassword.CheckedItems.Count == 0) return;//отмена генерации пароля при 0

string password = "";//пустой пароль

for (int i = 0; i < NUDPassLength.Value; i++)//цикл проверки значений

{

int n = rnd.Next(0, CLBPassword.CheckedItems.Count);//случайны выбор элементов из массива

string s = CLBPassword.CheckedItems[n].ToString();//запись выбранного элемента в переменную S

switch (s)

{

case "Цифры":

password += rnd.Next(10).ToString();//запись случайных цифр (от 0 до 9)

break;

case "Прописные буквы":

password += Convert.ToChar(rnd.Next(65, 88));//преобразование чисел в символы

break;

case "Строчные буквы":

password += Convert.ToChar(rnd.Next(97, 122));//преобразование чисел в символы

break;

default:

password += spec\_chars[rnd.Next(spec\_chars.Length)];//запись случайных спец. символов

break;

}

TBPass.Text = password;//присвоение TextBox функции password

Clipboard.SetText(password);//копирование пароля в буфер обмена

}

}

private void comboBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void btnConvert\_Click(object sender, EventArgs e)

{

double m1 = metrica[CBFrom.Text];//получение данных из поля From по ключу из metrica

double m2 = metrica[CBTo.Text];

double n = Convert.ToDouble(TBFrom.Text);//получение и конвертирование числа n из поля TextBox

TBTo.Text = (n \* m1 / m2).ToString();//помещяем отношение чисел в TextBox и конвертируем его

}

private void btnSwap\_Click(object sender, EventArgs e)

{

string T = CBFrom.Text;//добавляем временную переменную с сохранение в ComboBox

CBFrom.Text = CBTo.Text;//Меняем свойства "Text" местами

CBTo.Text = T;//записываем в ComboBox переменную "t"

}

private void CBMetric\_SelectedIndexChanged(object sender, EventArgs e)

{

switch (CBMetric.Text)

{

case "Длина":

metrica.Clear();//очистка словаря

metrica.Add("mm", 1);//дополняем словарь

metrica.Add("cm", 10);

metrica.Add("dm", 100);

metrica.Add("m", 1000);

metrica.Add("km", 1000000);

metrica.Add("mile", 1609344);

CBFrom.Items.Clear();//Очищаем своство Items

CBFrom.Items.Add("mm");//Добавляем новые своства в Items

CBFrom.Items.Add("cm");

CBFrom.Items.Add("dm");

CBFrom.Items.Add("m");

CBFrom.Items.Add("km");

CBFrom.Items.Add("mile");

CBTo.Items.Clear();//Очищаем своство Items

CBTo.Items.Add("mm");//Добавляем новые своства в Items

CBTo.Items.Add("cm");

CBTo.Items.Add("dm");

CBTo.Items.Add("m");

CBTo.Items.Add("km");

CBTo.Items.Add("mile");

CBFrom.Text = "mm";//меняем названия по умолчанию

CBTo.Text = "mm";//меняем названия по умолчанию

break;

case "Вес":

metrica.Clear();//очистка словаря

metrica.Add("g", 1);//дополняем словарь

metrica.Add("kg", 1000);

metrica.Add("t", 1000000);

metrica.Add("lb", 453.6);

metrica.Add("oz", 283);

CBFrom.Items.Clear();//Очищаем своство Items

CBFrom.Items.Add("g");//Добавляем новые своства в Items

CBFrom.Items.Add("kg");

CBFrom.Items.Add("t");

CBFrom.Items.Add("lb");

CBFrom.Items.Add("oz");

CBTo.Items.Clear();//Очищаем своство Items

CBTo.Items.Add("g");//Добавляем новые своства в Items

CBTo.Items.Add("kg");

CBTo.Items.Add("t");

CBTo.Items.Add("lb");

CBTo.Items.Add("oz");

CBFrom.Text = "g";//меняем названия по умолчанию

CBTo.Text = "g";//меняем названия по умолчанию

break;

default:

break;

}

}

}

}

# рандомайзер.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MyUtil

{

public partial class MainForm : Form

{

int count = 0;

Random rnd;//обработчик рандом

public MainForm()

{

InitializeComponent();

rnd = new Random();

}

private void tsmiExit\_Click(object sender, EventArgs e)

{

this.Close(); //присвоение функции выход

}

private void tsmiAbout\_Click(object sender, EventArgs e)

{

MessageBox.Show("Программа содержит ряд небольших программ," +

"\n которые могут пригодиться в жизни.", "О программе");

//после запятой - название окна; \n - перенос на другую строчку

}

private void btnPlus\_Click(object sender, EventArgs e)

{

count++;//складывает в счетчике

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnMinus\_Click(object sender, EventArgs e)

{

count--;//вычитает из счетчика

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnReset\_Click(object sender, EventArgs e)

{

count = 0;//обнуляет счетчик

lblCount.Text = Convert.ToString(count);//преобразование в стринг, но другим способом

}

private void button1\_Click(object sender, EventArgs e)

{

int n;

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);//Создание случайного числа

lblRandom.Text = n.ToString();//Вывод случайного числа в label

if (cbRandom.Checked)//Метод для CheckBox отключение и включение повторения чисел

{

int i = 0;

while (tbRandom.Text.IndexOf(n.ToString()) != -1)//исключение повторяющихся чисел

{

n = rnd.Next(Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value) + 1);

i++;//увеличение счетчика на 1

if (i > 1000) break;//прерывание цикла при попытках больше 1000

}

if(i<=1000) tbRandom.AppendText(n + " \n");// сохранение числа в случае i <= 1000

}

else

tbRandom.AppendText(n + " \n");//Сохранение чисел в TextBox

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

private void numericUpDown1\_ValueChanged(object sender, EventArgs e)

{

}

private void numericUpDown2\_ValueChanged(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void btnRandomClear\_Click(object sender, EventArgs e)

{

tbRandom.Clear();//очистка TextBox

}

private void btnCopy\_Click(object sender, EventArgs e)

{

Clipboard.SetText(tbRandom.Text);//Копирование в буфер обмена

}

}

}

# счетчик.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MyUtil

{

public partial class MainForm : Form

{

int count = 0;

public MainForm()

{

InitializeComponent();

}

private void tsmiExit\_Click(object sender, EventArgs e)

{

this.Close(); //присвоение функции выход

}

private void tsmiAbout\_Click(object sender, EventArgs e)

{

MessageBox.Show("Программа содержит ряд небольших программ," +

"\n которые могут пригодиться в жизни.", "О программе");

//после запятой - название окна; \n - перенос на другую строчку

}

private void btnPlus\_Click(object sender, EventArgs e)

{

count++;//складывает в счетчике

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnMinus\_Click(object sender, EventArgs e)

{

count--;//вычитает из счетчика

lblCount.Text = count.ToString();//преобразование в стринг

}

private void btnReset\_Click(object sender, EventArgs e)

{

count = 0;//обнуляет счетчик

lblCount.Text = Convert.ToString(count);//преобразование в стринг, но другим способом

}

}

# Выводы

Целью являлась: для достижения поставленной цели были созданы следующие программы:

* Form1.cs
* Счетчик
* генератор случайных чисел
* блокнот
* генератор паролей
* конвертер.

# Список Литературы

1. <https://gb.ru/chapters/969>
2. <https://docs.microsoft.com/ru-ru/dotnet/csharp/>
3. https://metanit.com/sharp/