

Лабораторна робота №14. Проект MyUtilites. Конвертер

Тема: Windows Forms. Проект MyUtilites. Конвертер

Мета заняття: навчитись працювати з словником (Dictionary) для створення списку значень елемента ComboBox, напрацювати навички з обробником подій Click елементів Button та виводом інформації у елемент TextBox з допомогою Windows Forms мовою C# у середовищі Microsoft Visual Studio.

1. Теоретичні відомості

Словник зберігає об'єкти, які представляють пару ключ-значення. Кожен такий об'єкт є об'єктом структури **KeyValuePair** <TKey, TValue>. Завдяки властивостям **Key** та **Value** можемо отримати ключ та значення елемента в словнику.

Розглянемо на прикладі використання словників:

```
1  using System;
2  using System.Collections.Generic;
3
4  class MainClass {
5      public static void Main (string[] args) {
6          Console.WriteLine ("Hello World");
7          Dictionary<int, string> countries = new Dictionary<int, string>(5);
8          countries.Add(1, "Ukraine");
9          countries.Add(3, "Great Britain");
10         countries.Add(2, "USA");
11         countries.Add(4, "France");
12         countries.Add(5, "China");
13
14         foreach (KeyValuePair<int, string> keyValue in countries)
15         {
16             Console.WriteLine(keyValue.Key + " - " + keyValue.Value);
17         }
18
19         // отримання елемента по ключу
20         string country = countries[4];
21         // зміна об'єкту
22         countries[4] = "Spain";
23         // видалення по ключу
24         countries.Remove(2);
25
26     }
27 }
28
```

Клас словник також надає методи **Add** і **Remove** для додавання і видалення елементів. Тільки в разі словників в метод **Add** передаються два параметри: ключ та значення, а метод **Remove** видаляє не за індексом, а по ключу.

Так як в прикладі ключами є об'єкти типу **int**, а значеннями - об'єкти типу **string**, то словник буде зберігати об'єкти **KeyValuePair<int, string>**. У циклі **foreach** можемо отримати і витягти з них ключ та значення.

Крім того, можемо отримати окремо колекції ключів і значень словника:

```
1  using System;
2  using System.Collections.Generic;
3
4  class MainClass {
5      public static void Main (string[] args) {
6          Dictionary<char, string> people = new Dictionary<char, string>();
7          people.Add('b', "Bill");
8          people.Add('t', "Tom");
9          people.Add('j', "John");
10
11         foreach (KeyValuePair<char, string> keyValue in people)
12         {
13             //Перебір елементів словника (Ключ-значення)
14             Console.WriteLine(keyValue.Key + " - " + keyValue.Value);
15         }
16
17         // перебор ключів
18         foreach (char c in people.Keys)
19         {
20             Console.WriteLine(c);
21         }
22
23         // перебор по значенню
24         foreach (string p in people.Values)
25         {
26             Console.WriteLine(p);
27         }
28     }
29 }
30 }
```

Ініціювати словники можна двома способами:

```
Dictionary<string, string> countries = new Dictionary<string, string>
{
    {"Франція", "Париж"},
    {"Германія", "Берлін"},
    {"Великобританія", "Лондон"}
};
foreach(var pair in countries)
    Console.WriteLine("{0} - {1}", pair.Key, pair.Value);
```

```
Dictionary<string, string> countries = new Dictionary<string, string>
{
    ["Франція"] = "Париж",
    ["Германія"] = "Берлин",
    ["Великобританія"] = "Лондон"
};

foreach (var pair in countries)
    Console.WriteLine("{0} - {1}", pair.Key, pair.Value);
```

2. Практичне завдання



У процесі виконання завдань лабораторної роботи необхідно формувати набори тестових даних для перевірки правильності виконання програмного коду. Створений код і результати перевірки його роботи потрібно помістити у звіт. Тестувати роботу програми рекомендується після додання чи зміни кожного оператора виведення.

Завдання. Створити проект «Конвертер величин».

1. Запускаємо Visual Studio. Відкриваємо проект MyUtilites (продовження проекту, створеного у лабораторних роботах 10-13).
2. Додаємо ще одну сторінку Конвертер. Вибираємо елемент `tabControl1` та властивість `TabPage5` (рис. 1).

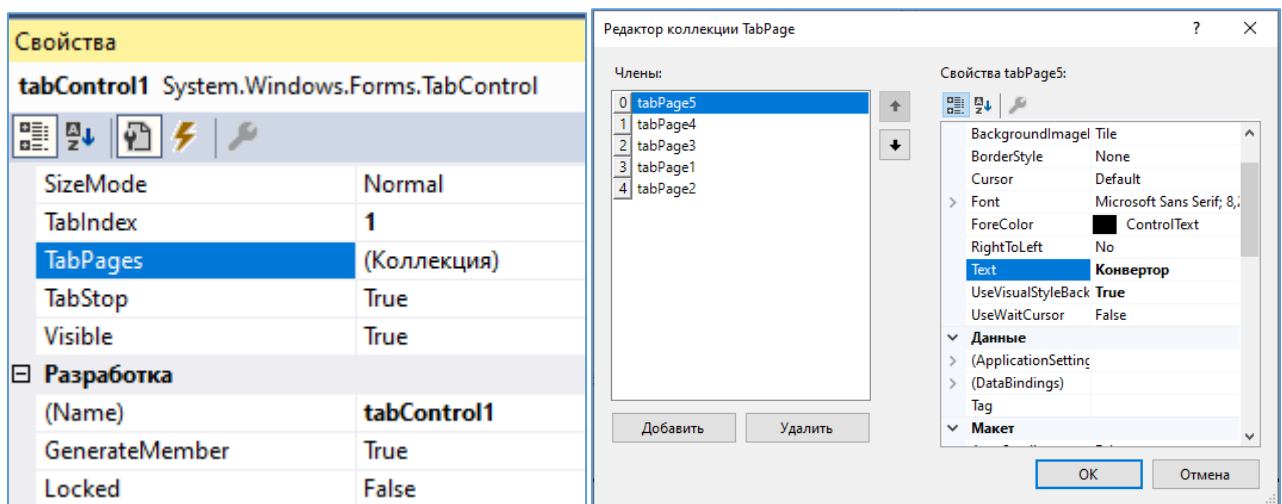


Рисунок 1 – Додавання TabPage5 (сторінка Конвертер) до елемента tabControl1

3. Додаємо на форму 2 елемента ComboBox, кнопку Button, 2 елемента TextBox (рис. 2).

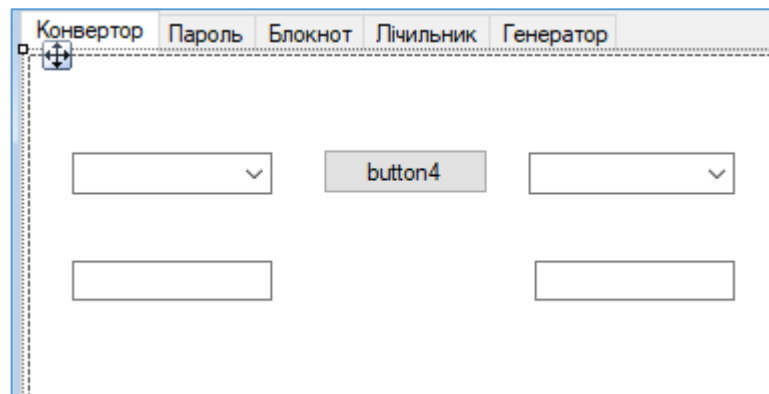


Рисунок 2 – Елементи форми. Додаток «Конвертер величин»

4. Встановити властивість елементів за таблицею 1.

Таблиця 1 – Властивості елементів форми. Сторінка «Конвертер»

Елемент	Поле	Значення
ComboBox1	Name	cbFrom
	Items	mm cm dm m kl mile
	Text	mm
ComboBox2	Name	cbTo
	Items	mm cm dm m kl mile
	Text	mm
Button	Name	btnConvert
	Text	Конвертувати
TextBox1	Name	tbFrom
	Text	1
TextBox1	Name	tbTo
	ReadOnly	True

5. Створюємо обробник події Click для кнопки. Створюємо елемент словник metrica (рис. 3)

Dictionary<string, double> metrica

```
namespace MyUtilites
{
    public partial class MainForm : Form
    {
        int count=0;
        Random rnd;
        char[] spec_char = new char[] { '%', '*', ')', '?', '#', '$', '^', '&', '~' };
        Dictionary<string, double> metrica;
```

Рисунок 3 – Оголошення словника metrica

6. Створюємо об'єкт metrica в конструкторі форм та заносимо дані в словник (рис. 4)

```
metrica = new Dictionary<string, double>();
metrica.Add("mm", 1);
metrica.Add("cm", 10);
metrica.Add("dm", 100);
metrica.Add("m", 1000);
metrica.Add("kl", 1000000);
metrica.Add("mile", 1609344);
```

```
public MainForm()
{
    InitializeComponent();
    rnd = new Random();
    metrica = new Dictionary<string, double>();
    metrica.Add("mm", 1);
    metrica.Add("cm", 10);
    metrica.Add("dm", 100);
    metrica.Add("m", 1000);
    metrica.Add("kl", 1000000);
    metrica.Add("mile", 1609344);
}
```

Рисунок 4 – Конструкторі форм. Створення об'єкт metrica

7. Додаємо код в обробник кнопки «Конвертувати» (рис. 5):

```
double m1 = metrica[cbFrom.Text];
double m2 = metrica[cbTo.Text];
double n = Convert.ToDouble(tbFrom.Text);
tbTo.Text = (n * m1 / m2).ToString();
```

```
private void btnConvert_Click(object sender, EventArgs e)
{
    double m1 = metrica[cbFrom.Text];
    double m2 = metrica[cbTo.Text];
    double n = Convert.ToDouble(tbFrom.Text);
    tbTo.Text = (n * m1 / m2).ToString();
}
```

Рисунок 5 – Код обробника подій Click для кнопки «Конвертувати»

8. Додаємо на форму кнопку зміни напрямку конвертації «<- ->» (рис. 6). Властивості Name присвоюємо значення btnSwap та створюємо для неї подію Click.

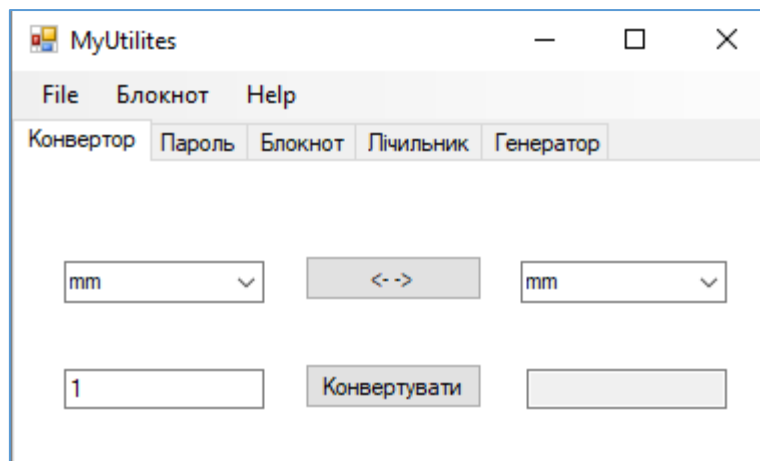


Рисунок 6 – Оновлений дизайн додатку «Конвертер величин»

```
private void btnSwap_Click(object sender, EventArgs e)
{
    string t = cbFrom.Text;
    cbFrom.Text = cbTo.Text;
    cbTo.Text = t;
}
```

9. Перевірте роботу програми «Конвертер величин» додатку MyUtilites (Ctrl+F5). Оформіть звіт.

3. Контрольні запитання

1. Як виконати ініціалізацію словника?
2. Як отримати доступ до елементів словника за ключом, значенням?
3. Як додати, видалити елемент до словника, як змінити значення словника?
4. Яка назва словника, що використовується у програмі? Який тип даних Key, Value у програмі?
5. Які дані знаходяться у словнику, що використовується у програмі лабораторної роботи?
6. Як виконана ініціалізація елементів словника у програмі? Як можна змінити код програми для ініціалізація елементів словника іншим методом.
7. Які зміни потрібно зробити у коді програми, щоб зробити додаток «Конвертор валют»?

Література

1. Евдокимов П. В. С# на примерах. СПб.: Наука и Техника, 2019. 320 с.
2. Маки А. Введение в .NET 4.0 и Visual Studio 2010 для профессионалов; пер. с англ. М. : ООО ИД "Вильямс", 2010. 416 с
3. С# 7.0. Справочник. Полное описание языка.: Пер. с англ. СПб.: ООО "Альфа-книга", 2018. 1024 с.
4. Троелсен, Эндрю, Джепикс, Филипп. Язык программирования С# 7 и платформы .NET и .NET Core. СПб. : ООО "Диалектика", 2018. 1328 с.
5. Офіційний сайт компанії Microsoft щодо технологій WPF та Windows Forms [Електронний ресурс]. – Режим доступу : <http://window-sclient.net>.
6. С#. Теорія та практика. URL: https://www.bestprog.net/uk/sitemap_ua/c-3
7. Сажин А. Справочник по языку программирования С#. URL: <https://brainoteka.com/blogs/c-spravochnik>.
8. С# Tutorial URL <https://www.theengineeringprojects.com>.
9. Уроки С#. URL: <https://itproger.com/course/csharp>.
10. Полное руководство по С# 8 и .NET Core. URL: <https://metanit.com/sharp/>
11. С#. Мини-программы. URL: <https://geekbrains.ru/chapters/969>