

## Лабораторна робота №10. Проект MyUtilites. Робота з елементом TabControl. Лічильник

**Тема:** Windows Forms. Проект MyUtilites. Робота з елементом TabControl.

**Мета заняття:** навчитись працювати з елементом TabControl для створення вкладок додатку, напрацювати навички з обробником подій Click елементів Button та виводом цифрової інформації з допомогою Windows Forms мовою C# у середовищі Microsoft Visual Studio.

### 1. Теоретичні відомості

Елемент TabControl дозволяє створити елемент керування з декількома вкладками. Кожна вкладка зберігає деякий набір різних елементів управління, як кнопки, текстові поля і ін. Кожна вкладка представлена класом TabPage .

Щоб налаштувати вкладки елемента TabControl використовуємо властивість TabPages. При перенесенні елемента TabControl з панелі інструментів на форму за замовчуванням створюються дві вкладки - tabPage1 і tabPage2. Змінимо їх відображення за допомогою властивості TabPages:

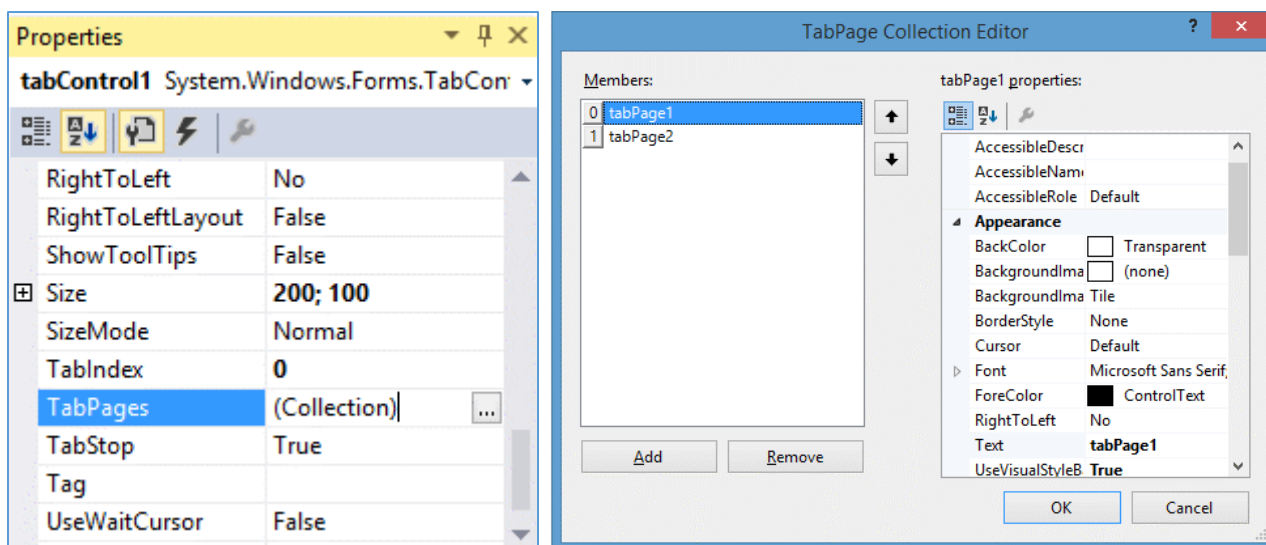


Рисунок 1 – Властивість TabPages елемента TabControl

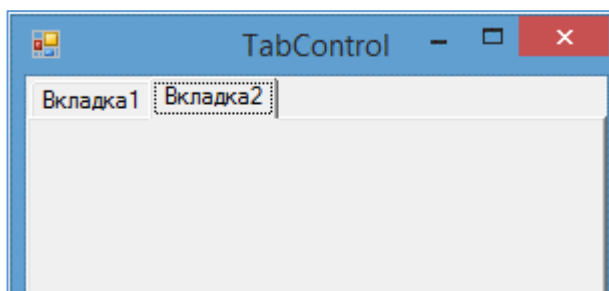


Рисунок 1.2 – Зовнішній вигляд вкладок на формі

Кожна вкладка представляє свого роду панель (рис. 1.2), на яку можемо додати інші елементи управління, а також заголовок, за допомогою якого можемо перемикатися по вкладках. Текст заголовка задається за допомогою властивості **Text**.

Для додавання нової вкладки можна її програмно створити і додати в колекцію **tabControl1.TabPages** за допомогою методу **Add**:

```
//додавання вкладки
TabPage newTabPage = new TabPage();
newTabPage.Text = "Розрахунок";
tabControl1.TabPages.Add(newTabPage);
```

Видалення теж просто:

```
// видалення вкладки за індексом
tabControl1.TabPages.RemoveAt(0);
// видалення вкладки за об'єктом
tabControl1.TabPages.Remove(newTabPage);
```

Отримавши в колекції **tabControl1.TabPages** потрібну вкладку за індексом, можемо їй задавати властивості:

```
// налаштування властивостей
tabControl1.TabPages[0].Text = "Перша вкладка";
```

## 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 1. Створити проект для розв'язання задачі.

*Створити меню **File, Help** додатку **MyUtilites**.*

1. Запускаємо Visual Studio. Відкриваємо проект MyUtilites (продовження проекту, створеного у лабораторній роботі 9).

2. На панелі елементів вибираємо елемент TabControl і перетаскуємо його на форму проекту (рис. 2.1). Властивості Dock встановлюємо значення Fill (рис. 2.2, а).

3. Вибираємо властивість TabPages «Collection» (рис. 2.2, b). У вікні, що відкрилось, міняємо властивість Text у TabPage на «Лічильник» (рис. 2.3).

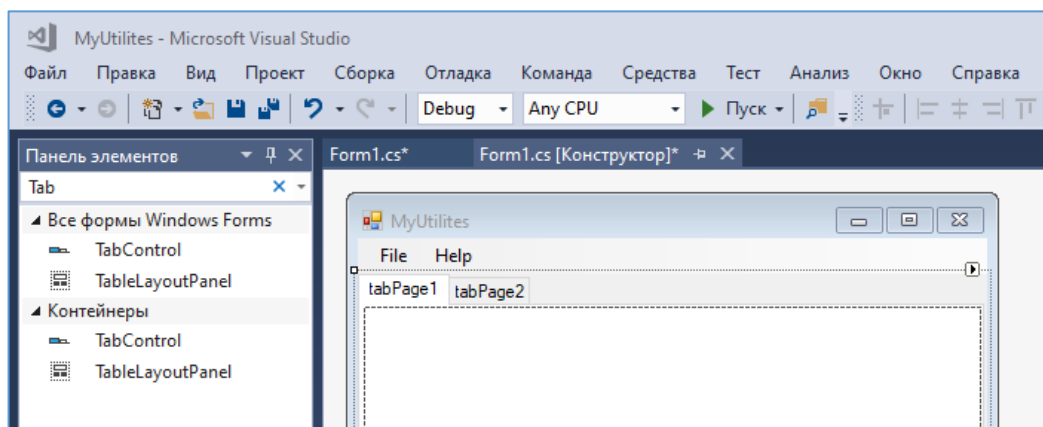
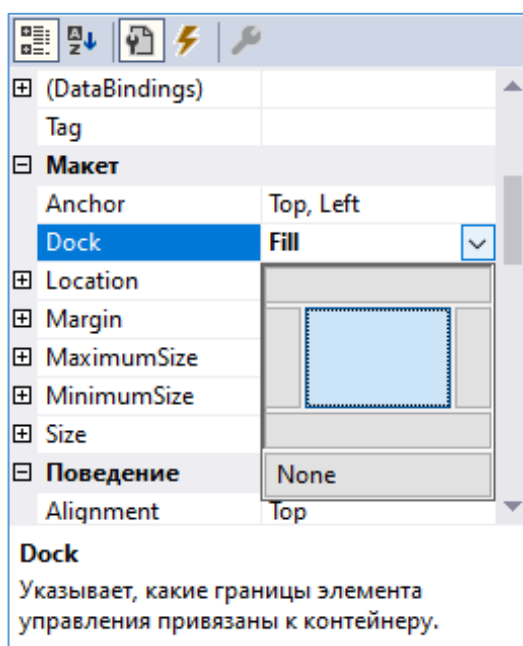
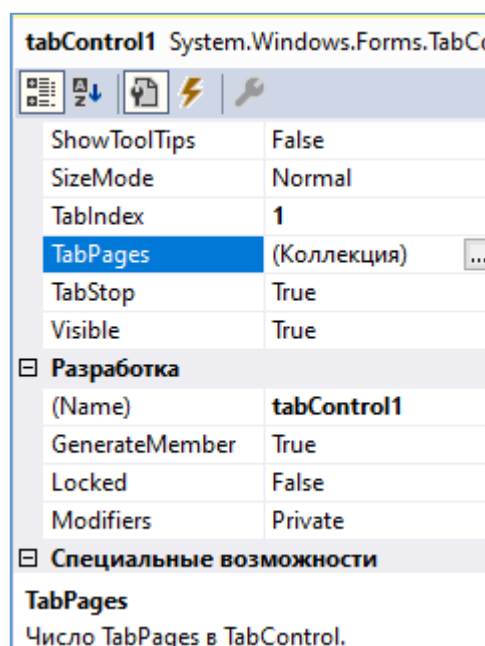


Рисунок 2.1 – Панель керування. Вибір елементу TabControl



a)



b)

Рисунок 2.2 – Налаштування властивості Dock та Collection TabControl

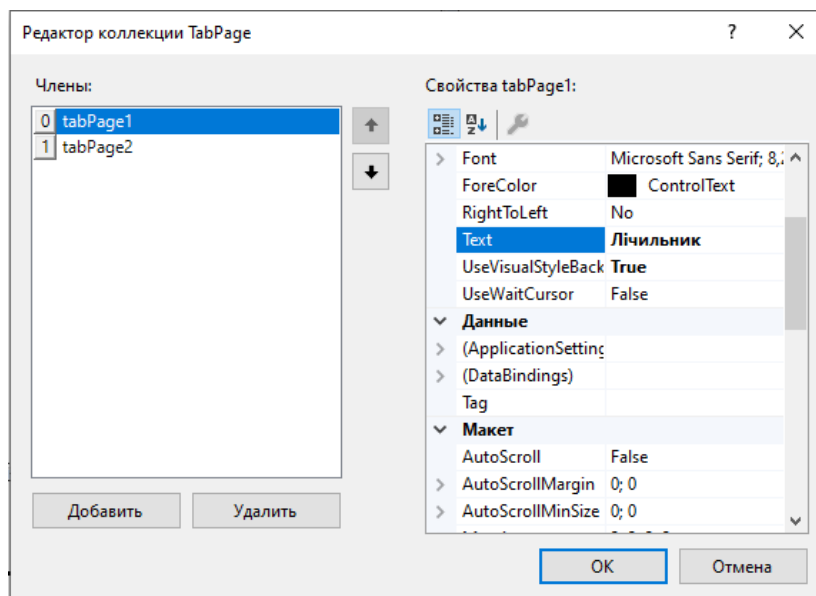


Рисунок 2.3 – Встановлення значення властивості Text елемента TabPage

4. До проекту додаємо 3 кнопки з назвами («+», «-», «Reset») та мітку Label 1 (рис. 2.4).

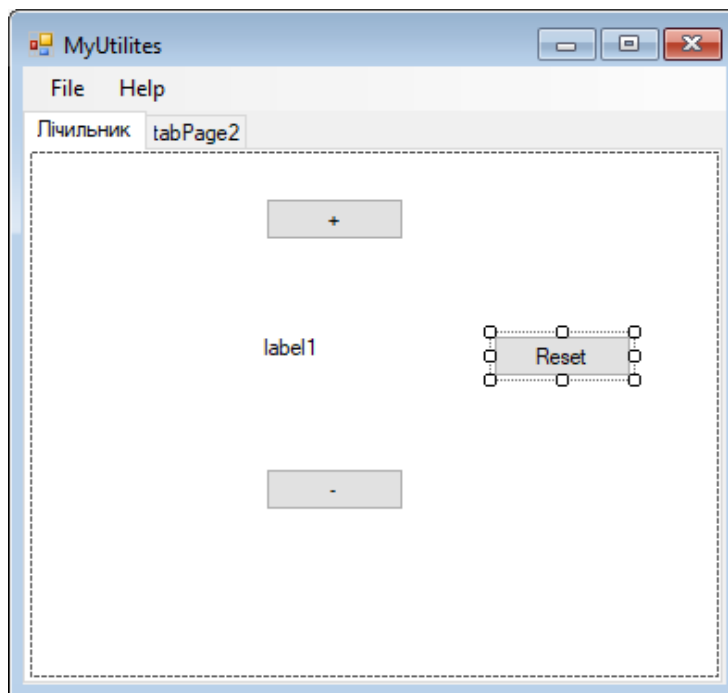


Рисунок 2.4 – Додавання до проекту елементів Button, Label

5. Властивість кнопок Name змінюємо на btnPlus, btnMinus, btnReset. Властивість Name мітки змінюємо на lblCount, а властивість Text на 0. Для кнопок та мітки встановлюємо властивість Font 14.

6. У конструкторі коду додаємо `int count=0` (рис. 2.5)

```
namespace MyUtilites
{
    public partial class MainForm : Form
    {
        int count=0;

        public MainForm()
        {
```

Рисунок 2.5 – Додавання змінної count типу int

7. Додаємо обробник подій для кнопки «+» (рис. 2.6)

```
private void button1_Click(object sender, EventArgs e)
{
    count++;
    lblCount.Text = count.ToString();
}
```

Рисунок 2.6 – Код обробника події Click кнопки btnPlus

8. Додаємо обробник подій для кнопки «-» (рис. 2.7)

```
private void button2_Click(object sender, EventArgs e)
{
    count--;
    lblCount.Text = count.ToString();
}
```

Рисунок 2.7 – Код обробника події Click кнопки btnMinus

9. Додаємо обробник подій для кнопки «Reset» (рис. 2.8).

```
private void button3_Click(object sender, EventArgs e)
{
    count=0;
    lblCount.Text = Convert.ToString(count);
}
```

Рисунок 2.8 – Код обробника події Click кнопки btnReset

10. Перевірте роботу елементів меню додатку MyUtilites (Ctrl+F5).  
Оформіть звіт.

### 3. Контрольні запитання

1. Призначення елемента TabControl.
2. Як задати власну назву вкладки tabPage1?
3. Яке призначення властивості Dock та його значення Fill елемента TabControl?
4. Які додати декілька вкладок елемента TabControl ?
5. Як додати вкладку елемента TabControl у кодї програми та задати її назву?
6. Як користуватись класу Convert для перетворення типу даних?
7. Які існують способи перетворення числових даних у рядкові та навпаки?

### 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](https://www.bestprog.net/uk/sitemap_ua/c-3)

7. Сажин А. Справочник по языку программирования C#. URL: <https://brainoteka.com/blogs/c-spravochnik>.
8. C# Tutorial URL <https://www.theengineeringprojects.com>.
9. Уроки C#. URL: <https://itproger.com/course/csharp>.
10. Полное руководство по C# 8 и .NET Core. URL: <https://metanit.com/sharp/>
11. C#. Мини-программы. URL: <https://geekbrains.ru/chapters/971>