**ПРАКТИЧНА РОБОТА № 4**

**«ОБРОБКА ВИКЛЮЧЕНЬ В МОВІ C#»**

**Мета роботи**: засвоїти основи обробки виключень в мові C#, набути практичних навичок обробки виключень.

**Вправа 1.**

using System;

class ExcDemo1

{

public static void Main()

{

int[] nums = new int[4];

try

{

Console.WriteLine("Перед генеруванням виключення.");

for (int i = 0; i < 10; i++)

{

nums[i] = i;

Console.WriteLine("nums[{0}]: {1}", i, nums[i]);

}

Console.WriteLine("Цей текст не відображається.");

}

catch (IndexOutOfRangeException)

{

Console.WriteLine("Індекс поза діапазоном!");

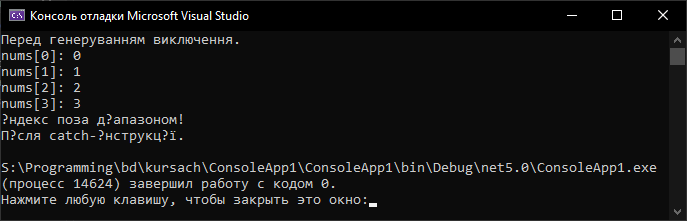
}

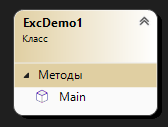
Console.WriteLine("Після catch-інструкції.");

}

}

Програма перехоплює вихід чисел за границю діапазона за допомогою обробника виняткових подій try-catch.





**Вправа 2.**

using System;

namespace AdvancedCSharp

{

public class MainEntryPoint

{

public static void Main()

{

string userInput; while (true)

{

try

{

Console.Write("Input а number between 0 and 5 " + "(or just hit return to exit)> ");

userInput = Console.ReadLine();

if (userInput == "") break;

int index = Convert.ToInt32(userInput);

if (index < 0 || index > 5)

throw new IndexOutOfRangeException("You typed in " + userInput); Console.WriteLine("Your number was " + index);

}

catch (IndexOutOfRangeException e)

{

Console.WriteLine("Exception: " + "Number should be between 0 and 5. " + e.Message);

}

catch (Exception e)

{

Console.WriteLine("An exception was thrown. Message was " + e.Message);

}

catch

{

Console.WriteLine("Some other exception has occurred");

}

finally

{

Console.WriteLine("Thank you");

}

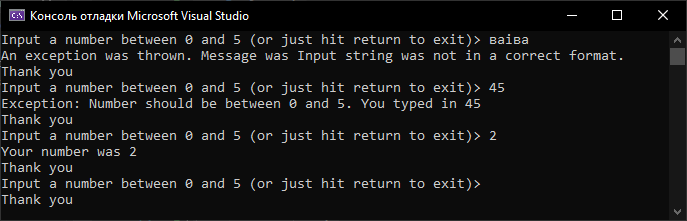
}

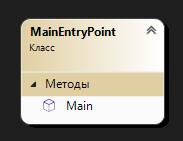
}

}

}

За допомогою блоків *try-catch-finally* програма перевіряє, чи входить у допустимий діапазон число, яке ввів користувач. Якщо не входить, виконається один з блоків *catch*.

****

****

**Завдання.**

Розробити простий калькулятор, ввівши в нього обробку виключень, пов’язаних з неправильним введенням даних.

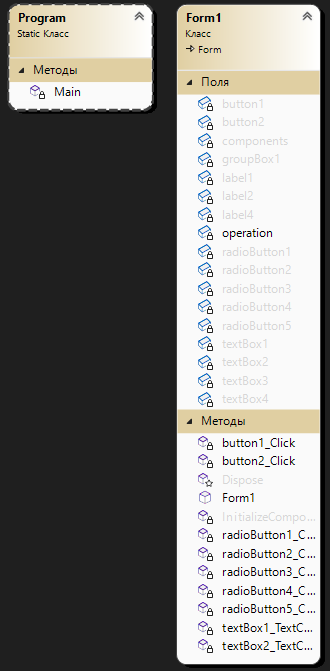
* 1. Лістинг *Form1.cs*:

1. using System;
2. using System.Collections.Generic;
3. using System.ComponentModel;
4. using System.Data;
5. using System.Drawing;
6. using System.Linq;
7. using System.Text;
8. using System.Threading.Tasks;
9. using System.Windows.Forms;
10. namespace prac4\_wf
11. {
12. public partial class Form1 : Form
13. {
14. string operation = "(pick operation)";
15. public Form1()
16. {
17. InitializeComponent();
18. }
19. private void button2\_Click(object sender, EventArgs e)
20. {
21. Application.Exit();
22. }

25. private void button1\_Click(object sender, EventArgs e)
26. {
27. try
28. {
29. if (radioButton1.Checked == true)
30. {
31. textBox4.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) + Convert.ToDouble(textBox2.Text));
32. }
33. else if (radioButton2.Checked == true)
34. {
35. textBox4.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) - Convert.ToDouble(textBox2.Text));
36. }
37. else if (radioButton3.Checked == true)
38. {
39. textBox4.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) \* Convert.ToDouble(textBox2.Text));
40. }
41. else if (radioButton4.Checked == true)
42. {
43. if(Convert.ToDouble(textBox2.Text) == 0)
44. {
45. textBox4.Text = "Divided by zero";
46. }
47. else
48. {
49. textBox4.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) / Convert.ToDouble(textBox2.Text));
50. }
52. }
53. else if (radioButton5.Checked == true)
54. {
55. textBox4.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) % Convert.ToDouble(textBox2.Text));
56. }
57. else
58. {
59. MessageBox.Show("Pick operation!");
60. }
61. }
62. catch (Exception ex)
63. {
64. MessageBox.Show(ex.Message);
65. }
66. }
67. private void textBox2\_TextChanged(object sender, EventArgs e)
68. {
69. textBox3.Text = "";
70. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
71. }
72. private void textBox1\_TextChanged(object sender, EventArgs e)
73. {
74. textBox3.Text = "";
75. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
76. }
77. private void radioButton1\_Click(object sender, EventArgs e)
78. {
79. operation = "+";
80. textBox3.Text = "";
81. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
82. }
83. private void radioButton2\_Click(object sender, EventArgs e)
84. {
85. operation = "-";
86. textBox3.Text = "";
87. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
88. }
89. private void radioButton3\_Click(object sender, EventArgs e)
90. {
91. operation = "\*";
92. textBox3.Text = "";
93. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
94. }
95. private void radioButton4\_Click(object sender, EventArgs e)
96. {
97. operation = "/";
98. textBox3.Text = "";
99. textBox3.Text = textBox1.Text + " " + operation+ " " + textBox2.Text;
100. }
101. private void radioButton5\_Click(object sender, EventArgs e)
102. {
103. operation = "%";
104. textBox3.Text = "";
105. textBox3.Text = textBox1.Text + " " + operation + " " + textBox2.Text;
106. }
107. }
108. }

Клас *Form1,* що є класом-наслідником від базового класу *Form* має опис методів для виконання дій над числами в калькуляторі. Методи radioButton1\_Click, radioButton2\_Click, radioButton3\_Click, radioButton4\_Click, radioButton5\_Click є описом перемикачів, що визначають, яка дія має виконуватись. Метод button1\_Click описує кнопку «Calculate» та має в собі обробник виняткових ситуацій *try-catch*. У блоці *try* описані дії, під час яких може виникнути помилка(конвертація отриманих даних та виконання арифметичних дій), у блоці *catch* описаний виклик повідомлення про помилку, що виникла.

UML-діаграма класів:



Демонстрація роботи:

