



# C#

## Введение и базовый синтаксис

### Урок 1

# Особенности C#



# Особенности C#

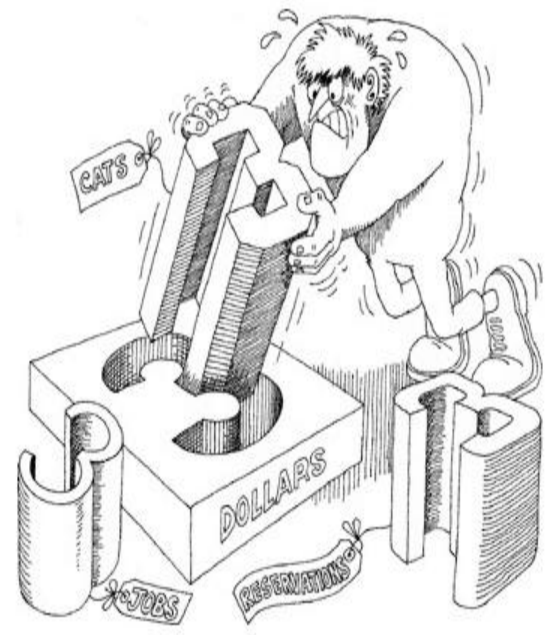
3

## 2) Обработка исключений

```
int[] a = new int[4];
try
{
    a[5] = 4; // тут возникнет исключение, так как у нас в массиве только 4 элемента
    Console.WriteLine("Завершение блока try");
}
catch (Exception ex)
{
    Console.WriteLine("Ошибка: " + ex.Message);
}
finally
{
    Console.WriteLine("Блок finally");
}
```

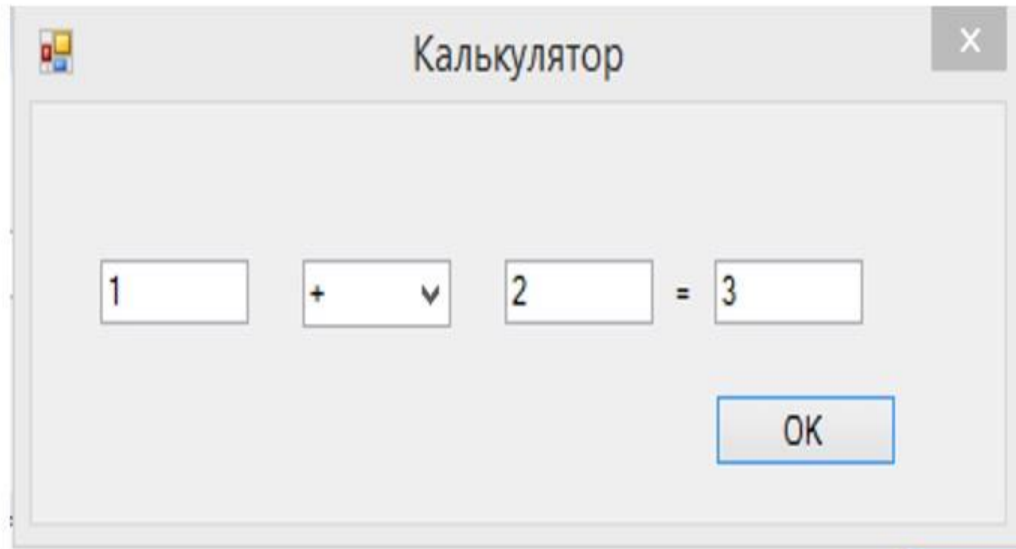
# Особенности C#

## 3) Строгая типизация



# Задание на урок

Написать программу «Калькулятор» и проверить её работу.

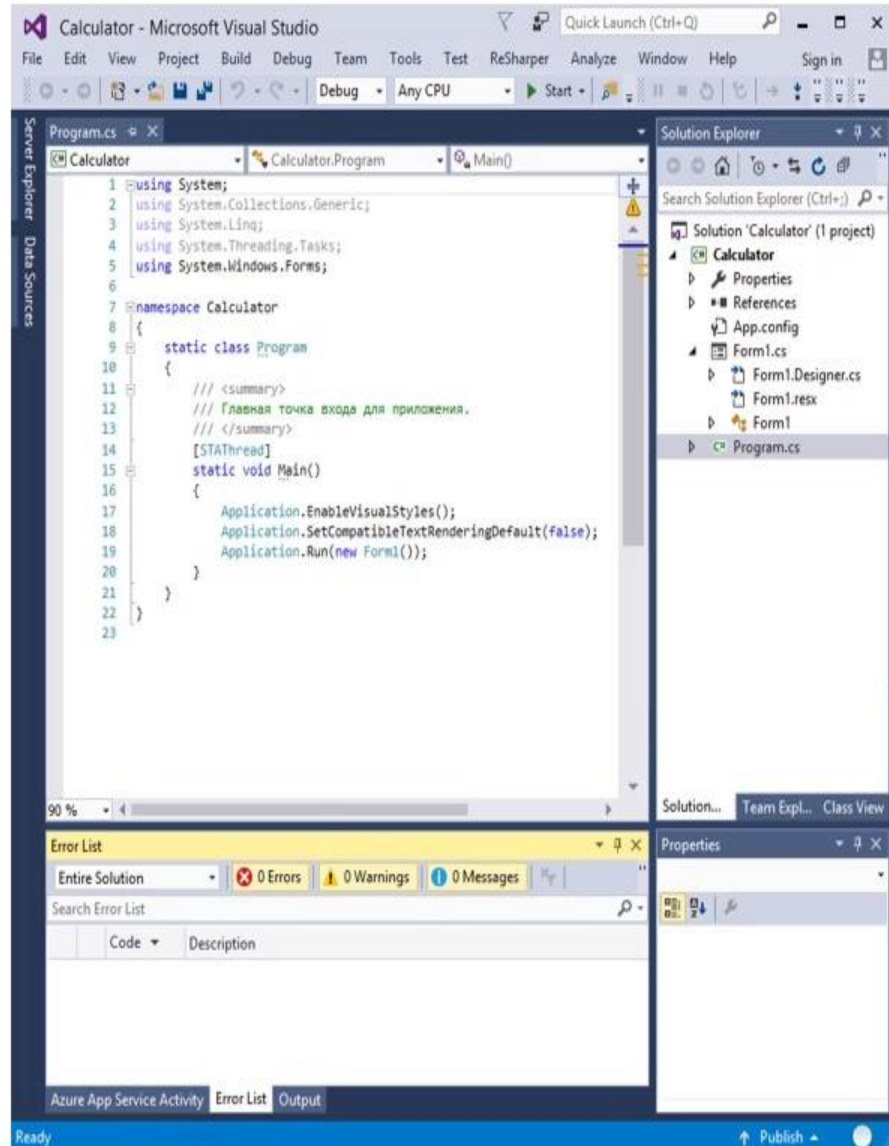


# Программа для разработки

6



Microsoft  
Visual Studio



# Создание проекта

7

Start Page

## Visual Studio

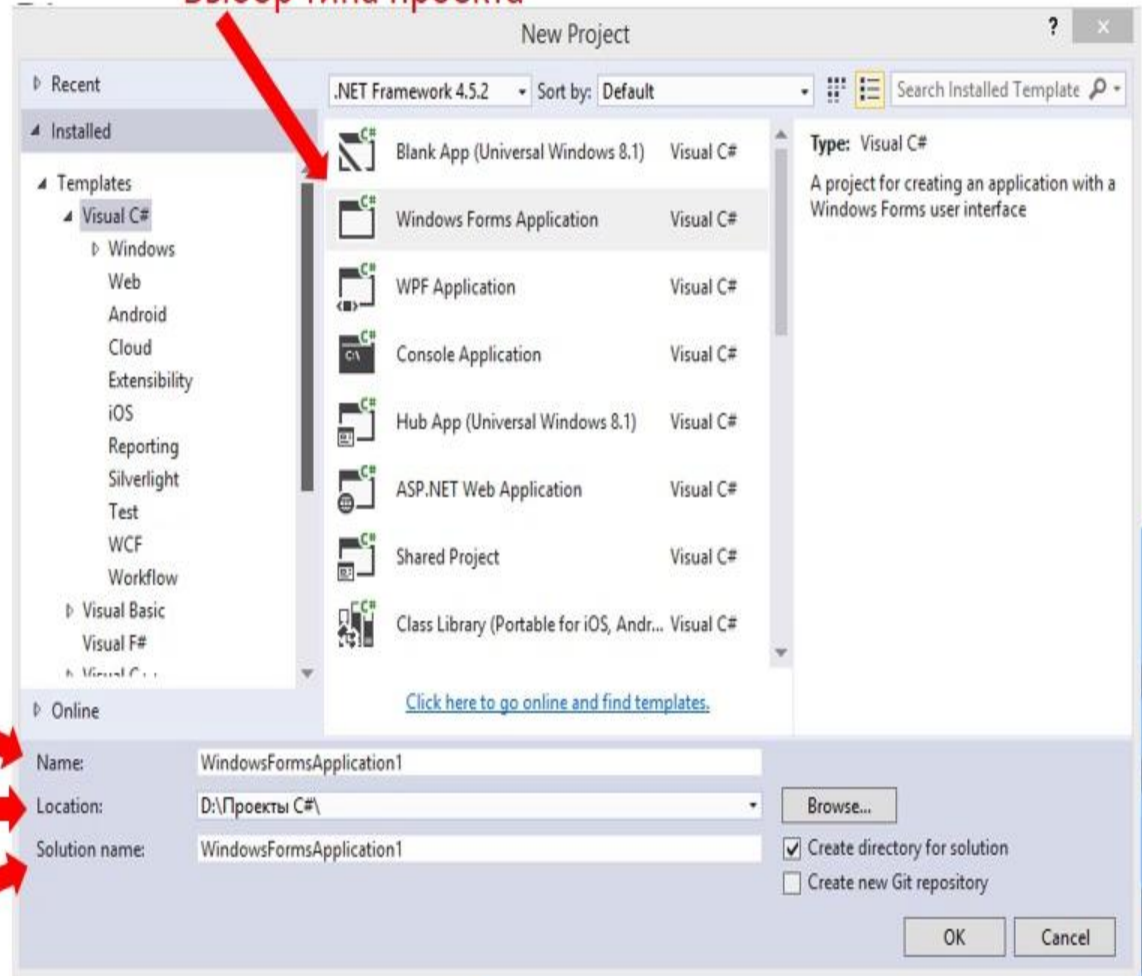
Start

New Project...

Open Project...

Open from Source Control...

Выбор типа проекта



Название проекта

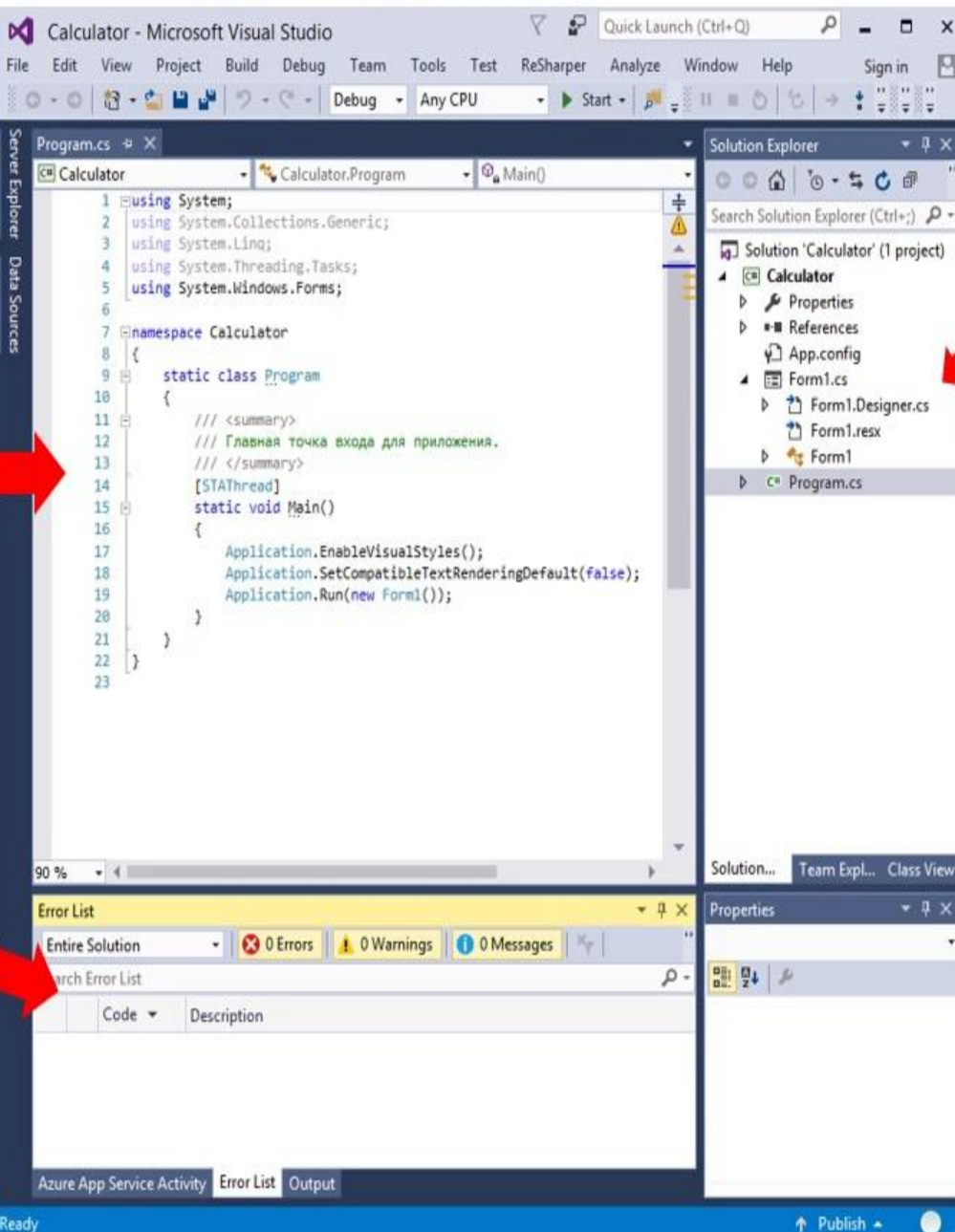
Путь к папке проекта

Название решения

# Структура Visual Studio

8

Редактор кода,  
в котором  
открыт файл  
Program.cs



Обозреватель решения  
(решение – несколько  
связанных между собой  
проектов)

Список  
обнаруженных  
ошибок



# Ошибка

9

В методе ошибка – он должен возвращать число (int)

```
20 public int SummOneAndTwo()  
21 {  
22     int i = 1 + 2;  
23 }  
24 }  
25 }
```

Code	Description	Project	File	Line	Suppression St...
CS0161	'Form1.SummOneAndTwo()': not all code paths return a value	WindowsFormsAp plication1	Form1.cs	20	Active

Код ошибки, описание, название проекта, файла и номер строки кода

# Работа с Windows Forms

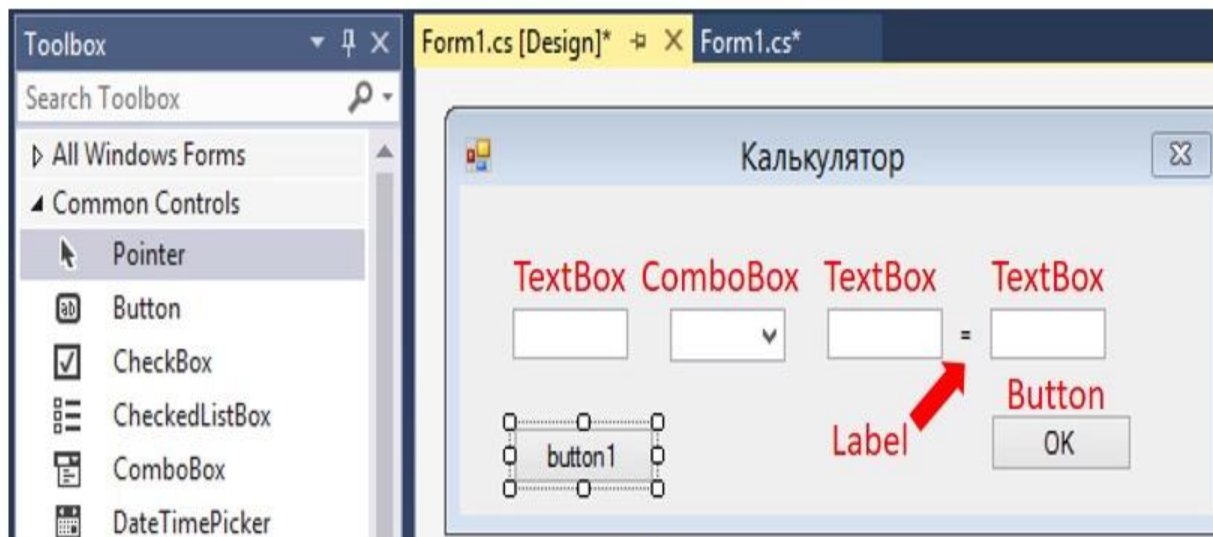
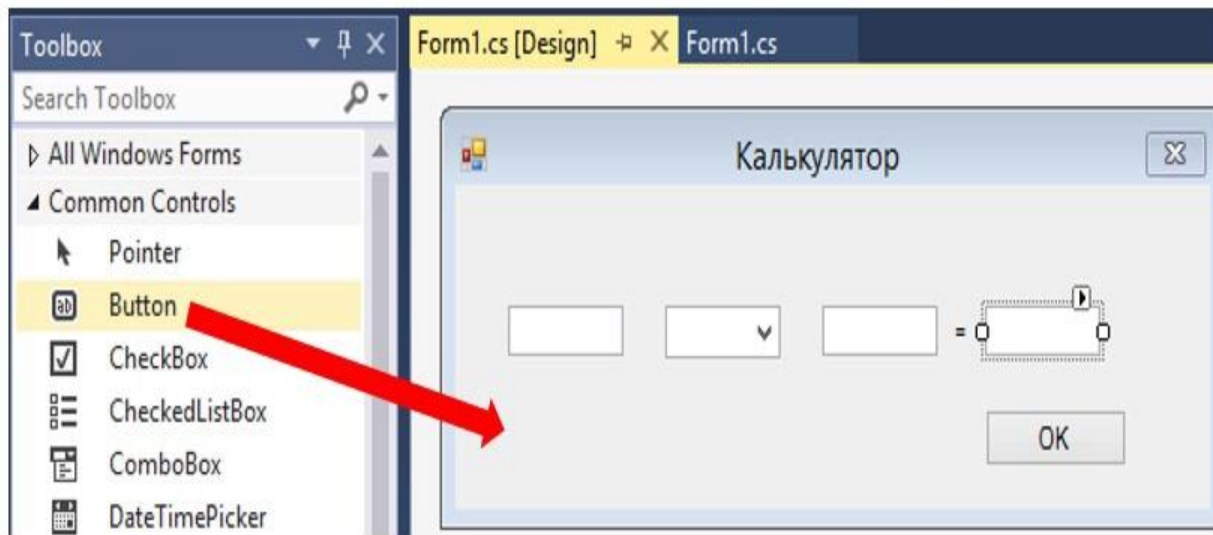
10

The screenshot displays the Visual Studio IDE with a Windows Forms application in design mode. The main window shows a blank form titled 'Form1'. The Solution Explorer on the right shows the project structure for 'WindowsFormsApplication1', including 'Form1.cs', 'Form1.Designer.cs', 'Form1.resx', 'Form1', and 'Program.cs'. The Error List at the bottom left shows 0 errors and 0 warnings. The Properties window at the bottom right shows the properties for 'Form1', including 'RightToLeftLayout', 'Text', 'UseWaitCursor', and 'Behavior'.

Code	Description

Property	Value
RightToLeftLayout	False
Text	Form1
UseWaitCursor	False
Behavior	
AllowDrop	False
AutoValidate	EnablePreventFocusCh

# Создаём интерфейс программы



# Создаём интерфейс программы

The image shows a screenshot of the Visual Studio IDE with the following components and annotations:

- Design View:** A window titled "Калькулятор" (Calculator) containing:
  - `textBoxOne`: A text input field on the left.
  - `textBoxTwo`: A text input field in the middle.
  - `comboBoxArithmetic`: A dropdown menu between the two text boxes.
  - `Выделенный элемент` (Selected element): A text box on the right, currently selected with a dashed border.
  - `buttonOK`: An "OK" button at the bottom.
- Solution Explorer:** Shows the project structure for "Calculator":
  - Calculator
    - Properties
    - References
    - App.config
    - Form1.cs
      - Form1.Designer.cs
      - Form1.resx
    - Form1
    - Program.cs
- Properties Window:** Shows the properties for the selected `textBoxResult` (System.Windows.Forms.TextBox):
 

Design	
(Name)	<code>textBoxResult</code>
GenerateMember	True
Locked	False

Red arrows point from the Russian annotations to the corresponding elements in the design and properties windows.

# Создаём интерфейс программы

13

The screenshot shows the Visual Studio IDE. In the foreground, a 'String Collection Editor' dialog box is open. It has a title bar with a question mark and a close button. Below the title bar is a 'Show output from:' field. The main area contains a text box with the instruction 'Enter the strings in the collection (one per line):'. Inside this text box, the characters '+' and '-' are entered on separate lines. At the bottom of the dialog are 'OK' and 'Cancel' buttons. A red arrow points from the text on the left to the '+' and '-' characters in the text box.

In the background, the 'Properties' window is visible for a control named 'comboBoxArithmetic'. The 'Items' property is set to '(Collection)'. A red arrow points from the text on the right to the '(Collection)' value.

DataSource	(none)
DisplayMember	(none)
Items	(Collection)
Tag	

Элементы, которые  
будут доступны для  
выбора в  
«ComboBox»

Одиночный  
клик

# Классы

Объект Волга класса Car

```
37 | Car volga = new Car("Волга", "Черный", 100, 200);
```

Класс Car (Автомобиль)

```
20 | public class Car
21 | {
22 |     private string brand;
23 |     private string color;
24 |     private int power;
25 |     private int maxSpeed;
26 |
27 |     public Car(string brand, string color, int power, int maxSpeed)
28 |     {
29 |         this.brand = brand;
30 |         this.color = color;
31 |         this.power = power;
32 |         this.maxSpeed = maxSpeed;
33 |     }
34 | }
```

Объект BMW класса Car

```
39 | Car bmw = new Car("BMW", "Белый", 150, 230);
```

# Класс Program

15

```
Program.cs  X Form1.Designer.cs  Form1.cs [Design]  Form1.cs
C# Calculator  Calculator.Program  Main()
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using System.Windows.Forms;
6
7  namespace Calculator
8  {
9      static class Program
10     {
11         /// <summary>
12         /// Главная точка входа для приложения.
13         /// </summary>
14         [STAThread]
15         static void Main()
16         {
17             Application.EnableVisualStyles();
18             Application.SetCompatibleTextRenderingDefault(false);
19             Application.Run(new Form1());
20         }
21     }
22 }
23
```

# Класс Form1

```
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
11 namespace WindowsFormsApplication1
12 {
13     public partial class Form1 : Form
14     {
15         public Form1()
16         {
17             InitializeComponent();
18         }
19     }
20 }
21
```



# Методы

Процедура (не возвращает значения)

```
1 static void Method1()  
2 {  
3     Console.WriteLine("Method1");  
4 }
```

Функция (возвращает строку)

```
6 string Hello()  
7 {  
8     return "Hell to World";  
9 }
```

# Методы

Тип  
возвращаемого  
значения

Название  
метода

Параметры метода

Уровень доступа

```
40 public int Summ(int first, int two) // метод, возвращающий сумму 2-ух чисел
41 {
42     int result = first + two;
43     return result;
44 }
```

# Обработчики событий

```
13 private void buttonOK_Click(object sender, EventArgs e)
14 {
15
16 }
```

# Взаимодействие с формой

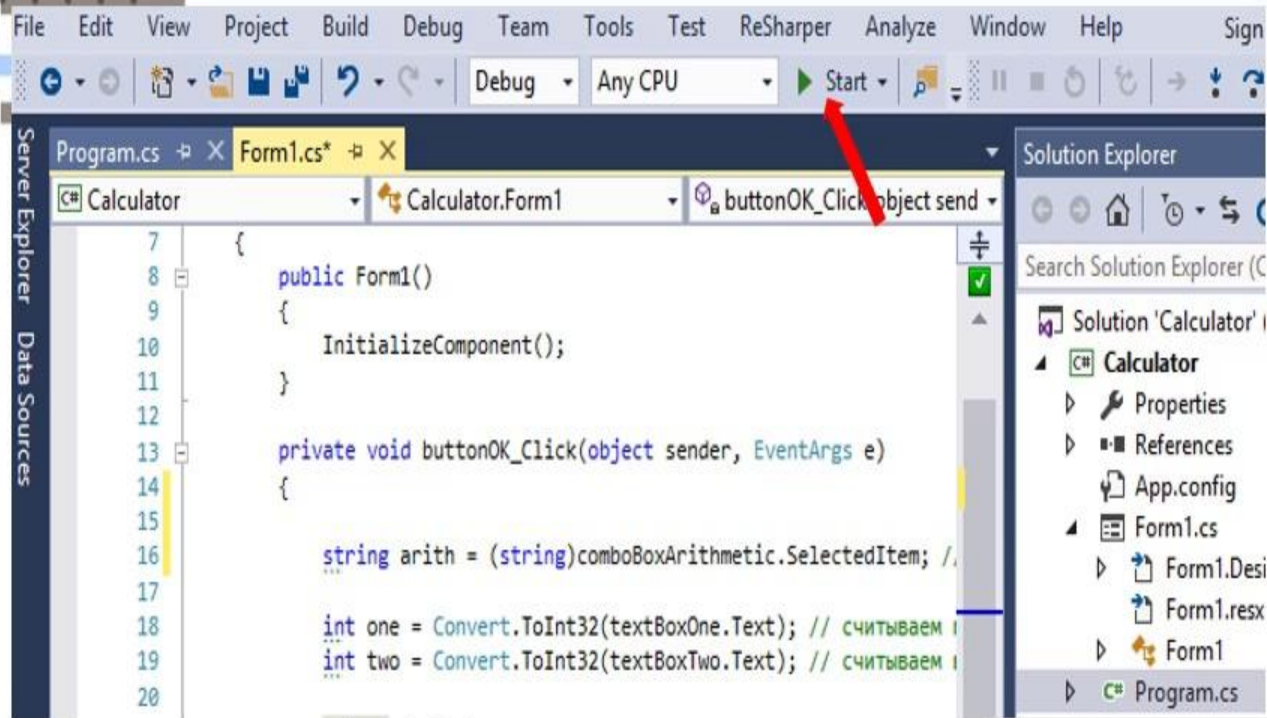
```
13 private void buttonOK_Click(object sender, EventArgs e)
14 {
15
16     string arith = (string)comboBoxArithmetic.SelectedItem; // считываем выбранную операцию
17                                                         // с использованием неявного преобразования
18     int one = Convert.ToInt32(textBoxOne.Text); // считываем первую переменную
19     int two = Convert.ToInt32(textBoxTwo.Text); // считываем вторую переменную
```

# Switch

```
21 switch (arith)
22 {
23     case "+":
24         string result = Summ(one, two).ToString(); // вызываем функцию сложения и конвертируем результат из числа в строку
25         textBoxResult.Text = result; // выводим результат в текстовое поле
26         break; // выход
27     case "-":
28         textBoxResult.Text = Subtraction(one, two).ToString(); // можно обойтись без дополнительной переменной для хранения результата
29         break;
30
31 }
```

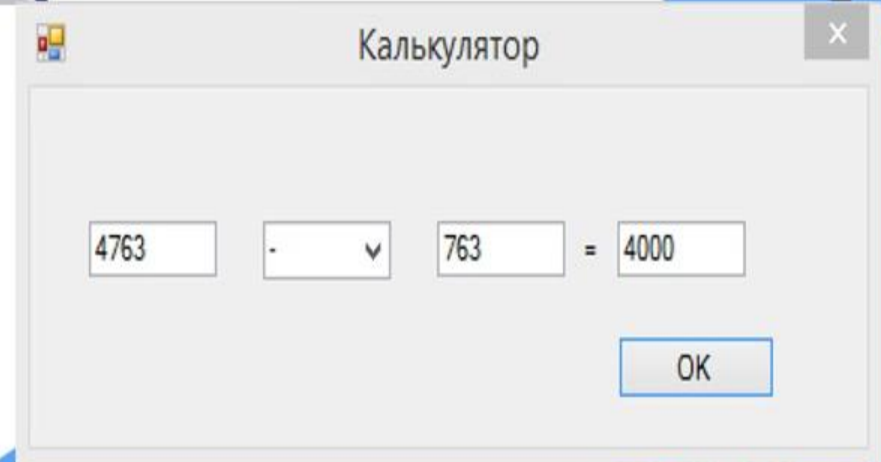
```
13 private void buttonOK_Click(object sender, EventArgs e)
14 {
15
16     string arith = (string)comboBoxArithmetic.SelectedItem; // считываем выбранную операцию
17                                     // с использованием неявного преобразования
18     int one = Convert.ToInt32(textBoxOne.Text); // считываем первую переменную
19     int two = Convert.ToInt32(textBoxTwo.Text); // считываем вторую переменную
20
21     switch (arith)
22     {
23         case "+":
24             string result = Summ(one, two).ToString(); // вызываем функцию сложения и конвертируем
25                                     // результат из числа в строку
26             textBoxResult.Text = result; // выводим результат в текстовое поле
27             break; // выход
28         case "-":
29             textBoxResult.Text = Subtraction(one, two).ToString(); // можно обойтись без дополнительной
30                                     // переменной для хранения результата
31             break;
32     }
33 }
34 public int Summ(int first, int two) // метод, возвращающий сумму 2-ух чисел
35 {
36     int result = first + two;
37     return result;
38 }
39 public int Subtraction(int first, int two) // метод, возвращающий разность 2-ух чисел
40 {
41     int result = first - two;
42     return result;
43 }
```

# Отладка приложения



The screenshot shows the Visual Studio IDE with the 'Start' button in the Debug toolbar highlighted by a red arrow. The menu bar includes File, Edit, View, Project, Build, Debug, Team, Tools, Test, ReSharper, Analyze, Window, Help, and Sign. The toolbar shows 'Debug' and 'Any CPU' dropdowns, followed by the 'Start' button. The Solution Explorer on the right shows the project structure for 'Calculator', including 'Form1.cs'.

```
7 {  
8     public Form1()  
9     {  
10        InitializeComponent();  
11    }  
12  
13    private void buttonOK_Click(object sender, EventArgs e)  
14    {  
15  
16        string arith = (string)comboBoxArithmetic.SelectedItem; //  
17  
18        int one = Convert.ToInt32(textBoxOne.Text); // считываем  
19        int two = Convert.ToInt32(textBoxTwo.Text); // считываем  
20
```



The screenshot shows a Windows calculator window titled 'Калькулятор'. The window contains two input fields with the numbers '4763' and '763', a subtraction operator '-' in a dropdown menu, an equals sign '=', and a result field containing '4000'. There is an 'OK' button at the bottom right.

4763	-	763	=	4000
OK				

# Задание

24

Доработать программу «Калькулятор»:

- Написать методы для деления и умножения 2-х целых чисел;
- Проверить работу методов в программе.



# Решение

25

```
13 private void buttonOK_Click(object sender, EventArgs e)
14 {
15
16     string arith = (string)comboBoxArithmetic.SelectedItem; // считываем выбранную операцию
17                                     // с использованием неявного преобразования
18     int one = Convert.ToInt32(textBoxOne.Text); // считываем первую переменную
19     int two = Convert.ToInt32(textBoxTwo.Text); // считываем вторую переменную
20
21     switch (arith)
22     {
23         case "+":
24             string result = Summ(one, two).ToString(); // вызываем функцию сложения и конвертируем
25                                                         // результат из числа в строку
26             textBoxResult.Text = result; // выводим результат в текстовое поле
27             break; // выход
28         case "-":
29             textBoxResult.Text = Subtraction(one, two).ToString(); // можно обойтись без дополнительной
30                                                                     // переменной для хранения результата
31             break;
32         case "*":
33             textBoxResult.Text = Multiplication(one, two).ToString();
34             break;
35         case "/":
36             textBoxResult.Text = Division(one, two).ToString();
37             break;
38     }
39
40     public int Summ(int first, int two) // метод, возвращающий сумму 2-ух чисел...
41     public int Subtraction(int first, int two) // метод, возвращающий разность 2-ух чисел...
42     public int Multiplication(int first, int two)
43     {
44         int result = first * two;
45         return result;
46     }
47     public int Division(int first, int two)
48     {
49         int result = first / two;
50         return result;
51     }
52 }
```