

Materi:

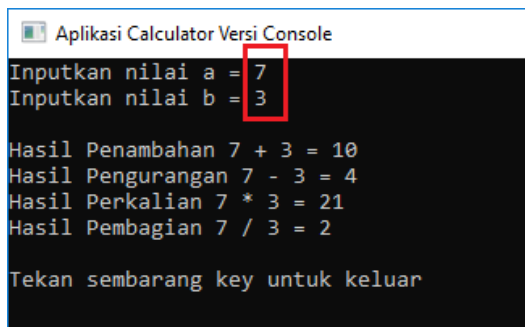
- ✓ Review IDE Visual Studio .NET dan Dasar-dasar Bahasa C#
- ✓ Bekerja dengan multi project
- ✓ Latihan Membuat Aplikasi Calculator Berbasis Console dan Windows Form

Pada praktikum pertama ini kita akan membuat aplikasi Calculator sederhana yang mempunyai empat method/operasi dasar yaitu :

- ✓ Penambahan
- ✓ Pengurangan
- ✓ Perkalian
- ✓ Pembagian

Tujuan dari praktikum ini adalah untuk mereview kembali beberapa materi Dasar-dasar Pemrograman yang sudah pernah Anda pelajari di semester sebelumnya. Untuk praktikum pertama ini, kita akan membuat dua aplikasi yang sama tapi dengan jenis project yang berbeda yaitu:

1. Aplikasi console

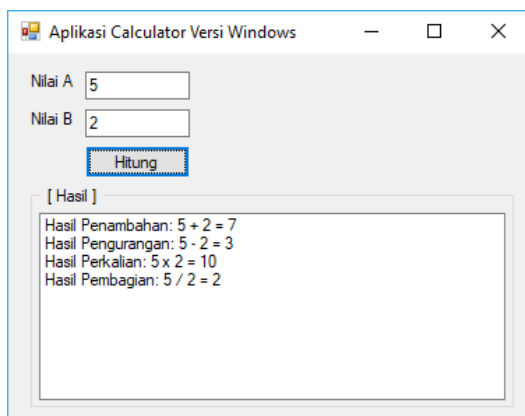


```
Aplikasi Calculator Versi Console
Inputkan nilai a = 7
Inputkan nilai b = 3

Hasil Penambahan 7 + 3 = 10
Hasil Pengurangan 7 - 3 = 4
Hasil Perkalian 7 * 3 = 21
Hasil Pembagian 7 / 3 = 2

Tekan sembarang key untuk keluar
```

2. Aplikasi windows



Aplikasi Calculator Versi Windows

Nilai A: 5
Nilai B: 2
Hitung

[Hasil]

Hasil Penambahan: 5 + 2 = 7
Hasil Pengurangan: 5 - 2 = 3
Hasil Perkalian: 5 x 2 = 10
Hasil Pembagian: 5 / 2 = 2

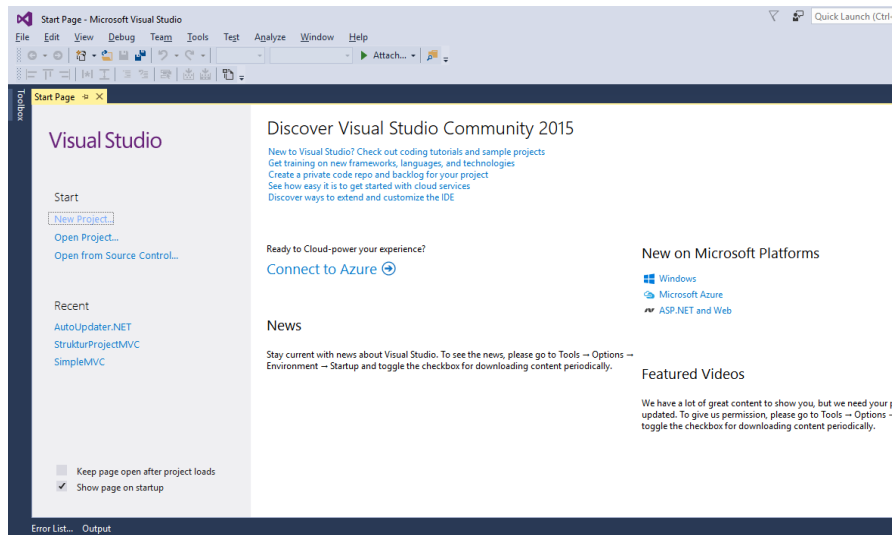
Pengenalan IDE Visual Studio .NET

Visual Studio .NET adalah sebuah **IDE (Integrated Development Environment)** yang merupakan software yang memiliki beberapa fasilitas yang diperlukan dalam pembangunan perangkat lunak.

Tujuan dari IDE adalah untuk menyediakan semua utilitas yang diperlukan dalam membangun perangkat lunak. Sebuah IDE, atau secara bebas dapat diterjemahkan sebagai Lingkungan Pengembangan Terpadu, setidaknya memiliki fasilitas:

- ✓ Editor, yaitu fasilitas untuk menuliskan kode sumber dari perangkat lunak.
- ✓ Compiler, yaitu fasilitas untuk mengecek sintaks dari kode sumber kemudian mengubah dalam bentuk binari yang sesuai dengan bahasa mesin.
- ✓ Linker, yaitu fasilitas untuk menyatukan data binari yang beberapa kode sumber yang dihasilkan compiler sehingga data-data binari tersebut menjadi satu kesatuan dan menjadi suatu program komputer yang siap dieksekusi.
- ✓ Debugger, yaitu fasilitas untuk mengetes jalannya program, untuk mencari *bug*/kesalahan yang terdapat dalam program.

IDE modern dapat membantu mempercepat pengembangan sebuah software. Pada saat penulisan kode, IDE juga dapat menunjukkan bagian-bagian yang jelas mengandung kesalahan atau keraguan.



Gambar IDE Visual Studio .NET

Pengenalan Bahasa C#

C# adalah Bahasa pemrograman baru yang diciptakan oleh Microsoft yang digunakan oleh banyak developer untuk mengembangkan aplikasi dengan platform .NET (.NET Framework). C# merupakan salah satu bahasa pemrograman berorientasi objek yang sintaknya berdasarkan bahasa C dan mirip dengan bahasa C++ dan Java.

Struktur Program C#

1. Namespace

Cara C# mengelompokkan sekumpulan types secara logic.

2. Using

Digunakan untuk menyederhanakan penulisan type declaration.

3. Type Declarations

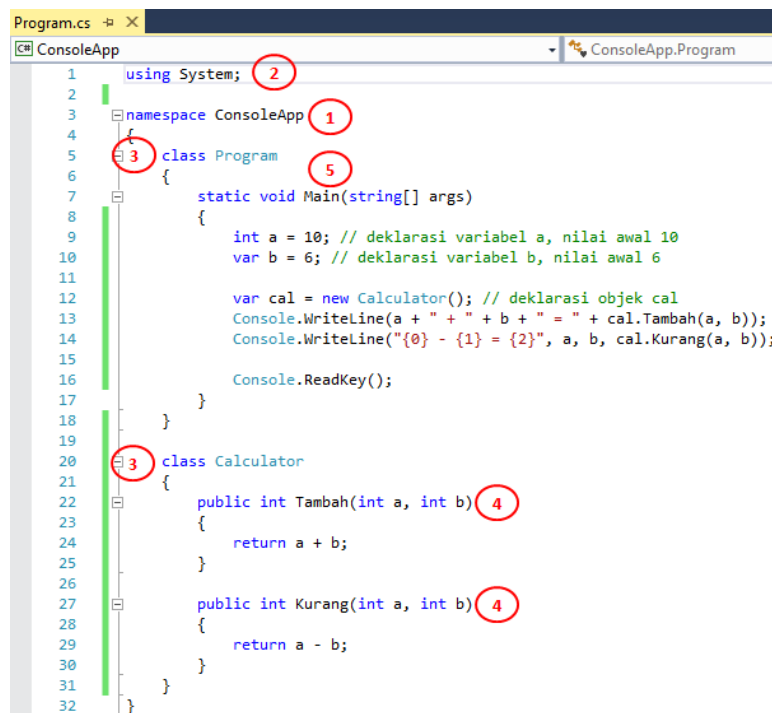
Terdiri dari class, interface, enum dan delegate.

4. Members

Terdiri dari const, field, method, properties, event, operator, constructor, destructor.

5. Main Method

Entry point dari program yang akan dijalankan pertama kali



```
1 using System;
2
3 namespace ConsoleApp
4 {
5     class Program
6     {
7         static void Main(string[] args)
8         {
9             int a = 10; // deklarasi variabel a, nilai awal 10
10            var b = 6; // deklarasi variabel b, nilai awal 6
11
12            var cal = new Calculator(); // deklarasi objek cal
13            Console.WriteLine(a + " + " + b + " = " + cal.Tambah(a, b));
14            Console.WriteLine("{0} - {1} = {2}", a, b, cal.Kurang(a, b));
15
16            Console.ReadKey();
17        }
18    }
19
20    class Calculator
21    {
22        public int Tambah(int a, int b)
23        {
24            return a + b;
25        }
26
27        public int Kurang(int a, int b)
28        {
29            return a - b;
30        }
31    }
32 }
```

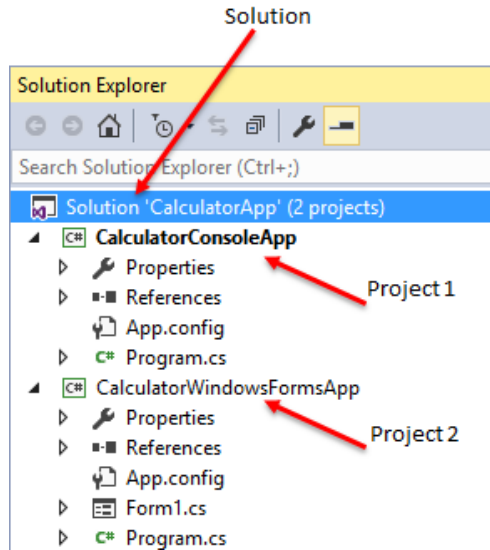
The image shows a screenshot of a C# program in Visual Studio. The code is annotated with red circles and numbers 1 through 5, corresponding to the sections in the preceding list:

- 1: `namespace ConsoleApp`
- 2: `using System;`
- 3: `class Program`
- 4: `public int Tambah(int a, int b)` and `public int Kurang(int a, int b)`
- 5: `static void Main(string[] args)`

Contoh kode program C#

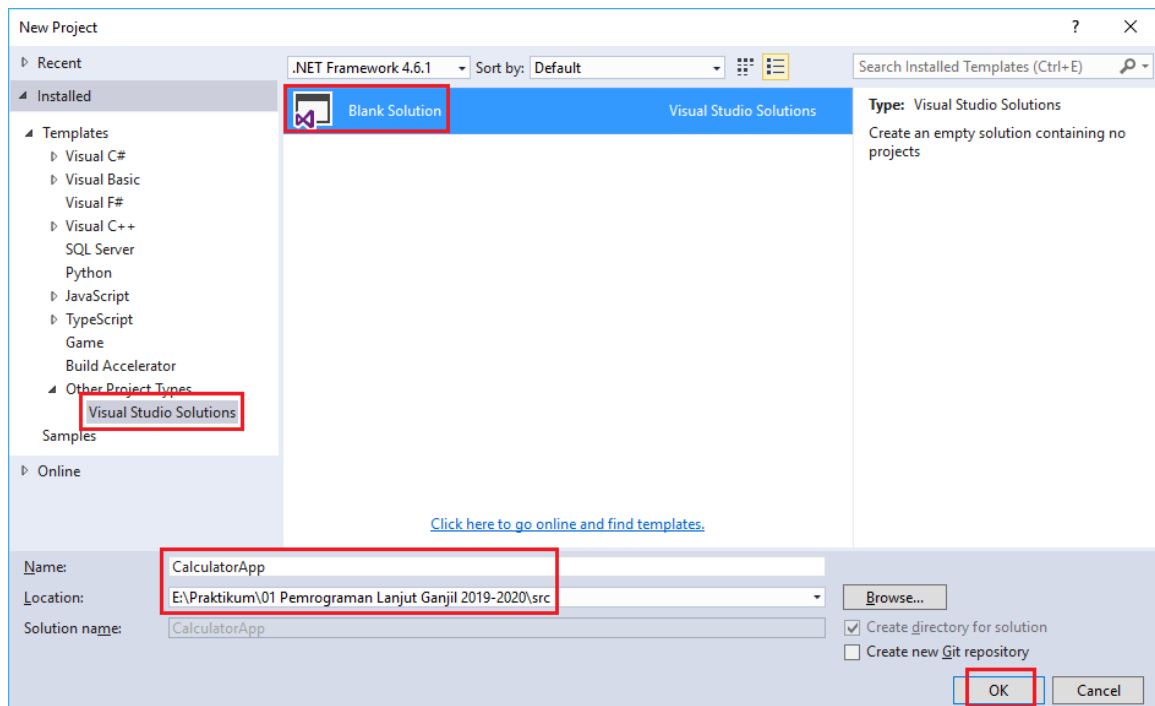
Langkah-langkah Membuat Solution

Pada praktikum ini kita akan bekerja dengan multi project, oleh karena itu kita akan membuat Solution terlebih dulu.



Langkah-langkahnya seperti berikut:

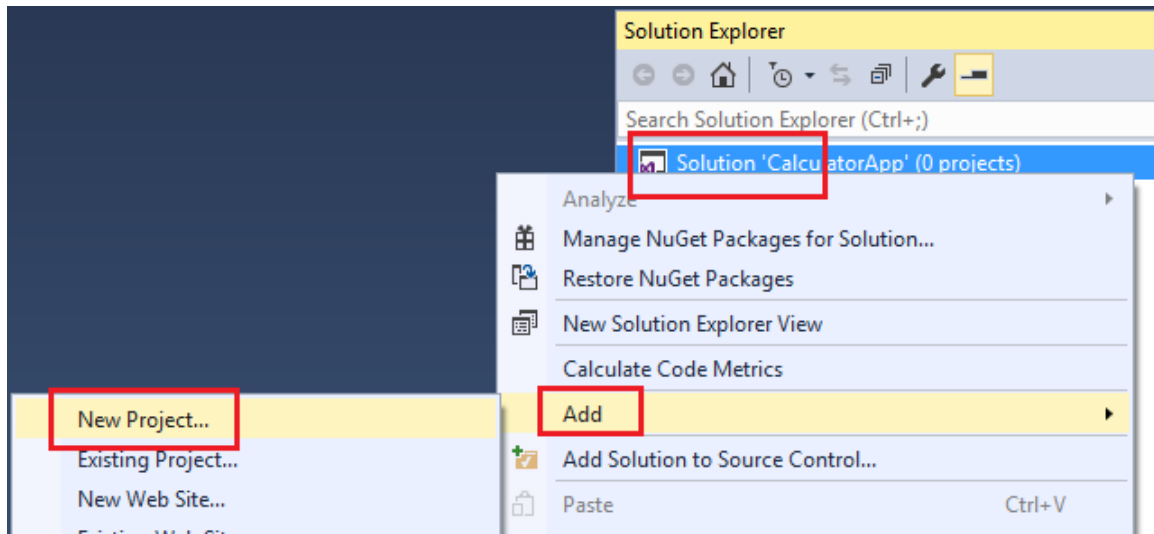
1. Jalankan Visual Studio .NET Community, kemudian klik menu File -> New -> Project. Pada panel *Templates* pilih *Other Project Types* -> *Visual Studio Solutions* -> *Blank Solution*



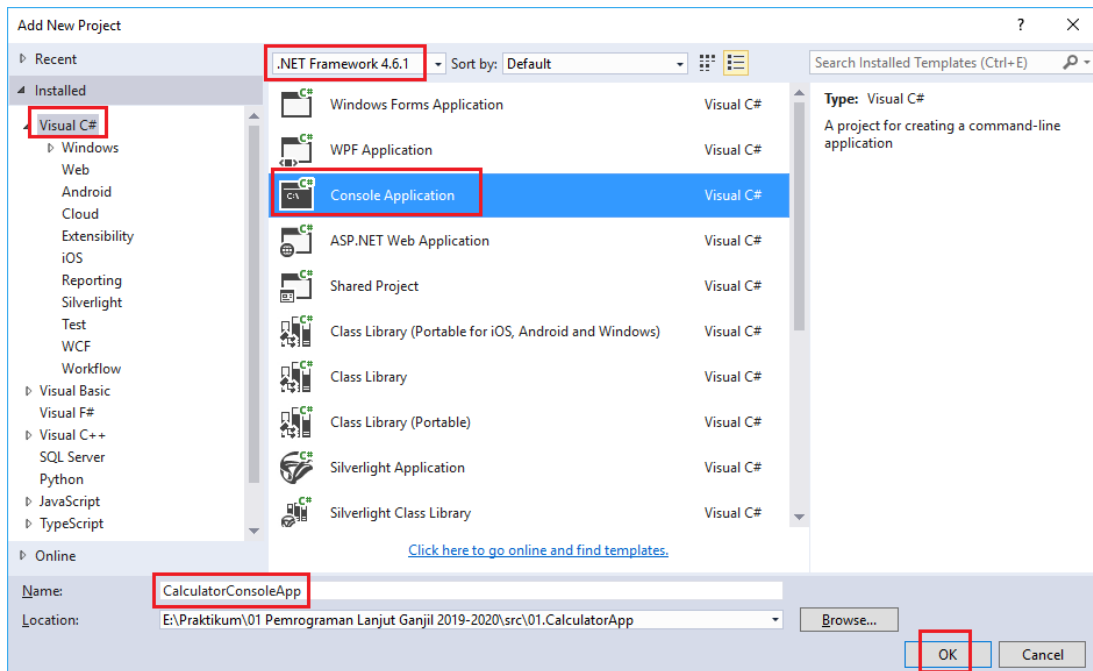
2. Untuk isian Name diisi dengan CalculatorApp dan untuk Location disesuaikan dengan lokasi folder Anda masing-masing.

Langkah-langkah Membuat Project

1. Setelah menambahkan *Solution* (langkah 1), langkah berikutnya adalah menambahkan project baru dengan cara klik kanan nama *Solution* -> *Add* -> *New Project*.

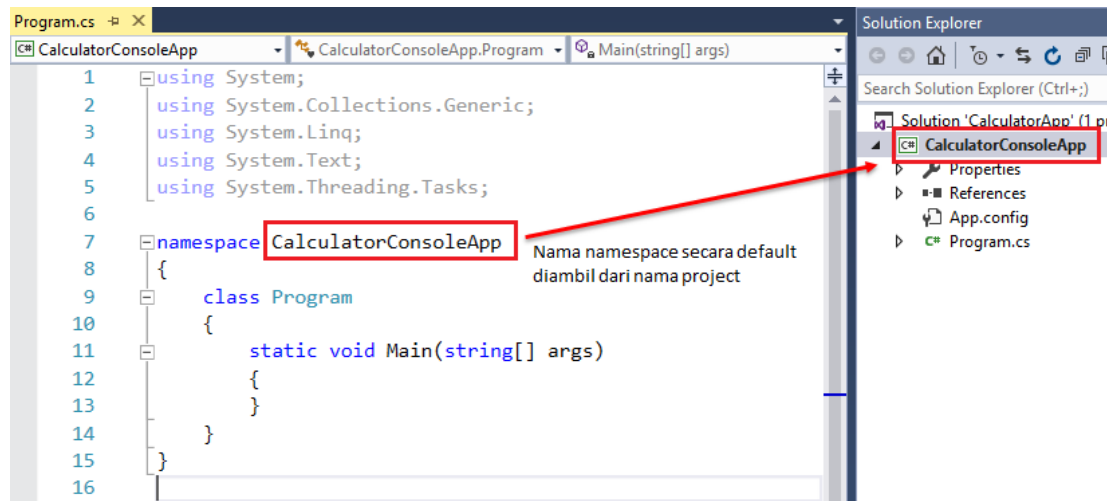


Kemudian pilih *ConsoleApplication*, untuk isian Name dan Location menyesuaikan.



Untuk isian Name diisikan dengan **CalculatorConsoleApp**. Untuk penamaan project perlu diperhatikan huruf besar kecilnya karena nama project secara default akan menjadi *namespace* yang akan berpengaruh terhadap penulisan kode program.

2. Apabila proses pembuatan project baru berhasil dibuat, maka akan tampil kode editor seperti berikut.



3. Langkah berikutnya kita akan menambahkan beberapa method/operasi yang berhubungan dengan fungsi dasar Calculator yaitu Penambahan, Pengurangan, Perkalian dan Pembagian. Berikut contoh method untuk Penambahan dan Pengurangan, sedangkan Perkalian dan Pembagian Anda tambahkan sendiri.

```
class Program
{
    static void Main(string[] args)
    {
    }

    static int Penambahan(int a, int b)
    {
        return a + b;
    }

    static int Pengurangan(int a, int b)
    {
        return a - b;
    }
}
```

```
// TODO: Anda tambahkan kode untuk method Perkalian dan Pembagian
}
```

Pada kode di atas kita menambahkan dua method masing-masing dengan nama *Penambahan* dan *Pengurangan*. Kedua method ini mempunyai parameter yang sama yaitu *a* dan *b* yang masing-masing bertipe *int*. Parameter ini berfungsi sebagai media input bagi sebuah method.

4. Setelah semua method calculatornya lengkap, kemudian Anda lengkapi kode untuk method Mainnya seperti berikut:

```
static void Main(string[] args)
{
    Console.Title = "Aplikasi Calculator Versi Console";

    Console.Write("Inputkan nilai a = ");
    int a = int.Parse(Console.ReadLine()); // casting

    Console.Write("Inputkan nilai b = ");
    var b = Convert.ToInt32(Console.ReadLine()); // casting

    Console.WriteLine(); // tambahkan baris kosong
    Console.WriteLine("Hasil Penambahan: {0} + {1} = {2}", a, b, Penambahan(a, b));
    Console.WriteLine("Hasil Pengurangan: {0} - {1} = {2}", a, b, Pengurangan(a,
b));
    Console.WriteLine("Hasil Perkalian: {0} x {1} = {2}", a, b, Perkalian(a, b));
    Console.WriteLine("Hasil Pembagian: {0} / {1} = {2}", a, b, Pembagian(a, b));

    Console.ReadKey();
}
```

Pada kode di atas method Main berfungsi sebagai entry poin yang akan dipanggil pertama kali oleh sistem operasi ketika program di jalankan. Di method *Main* ini kita memanggil semua method *Calculator* dan menampilkan hasilnya.

Untuk menampung inputan dari keyboard kita memanggil method *ReadLine* dari class *Console*. Method ini akan mengembalikan nilai berupa tipe data string, sehingga sebelum di simpan datanya ke variabel *a* yang bertipe integer harus dilakukan proses konversi/casting terlebih dulu dari tipe string ke integer.

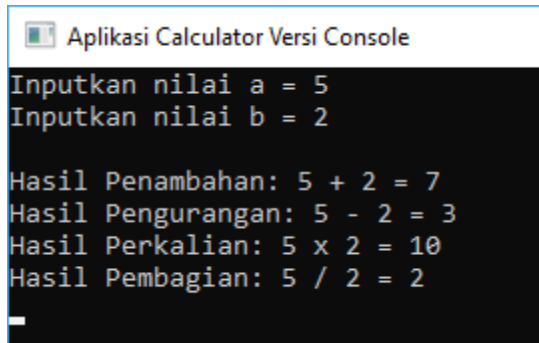
Tanda {0..3} adalah format string yang menunjukkan posisi index dari variabel atau objek yang akan ditampilkan nilainya.

```
Console.WriteLine("Hasil Penambahan: {0} + {1} = {2}", a, b, Penambahan(a, b));
```



Pada kode di atas, format string {0} akan digantikan nilainya dengan variabel *a*, {1} dengan nilai variable *b* dan seterusnya.

- Setelah menyelesaikan penulisan kode, untuk menjalankan program tersebut tekan tombol F5 (Start Debugging)



```
Aplikasi Calculator Versi Console
Inputkan nilai a = 5
Inputkan nilai b = 2

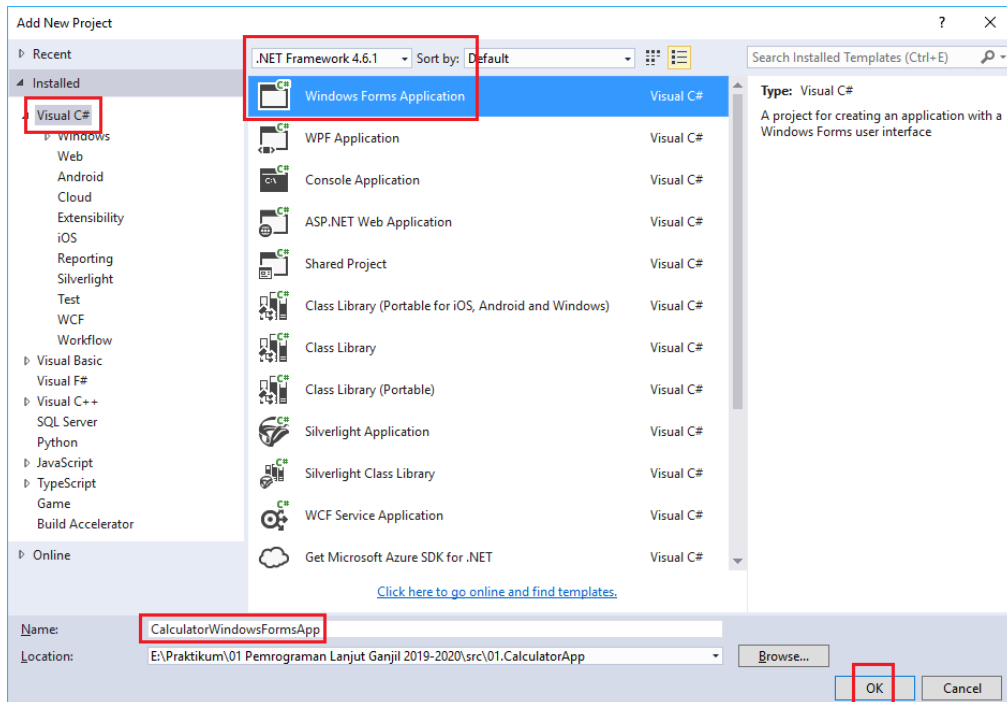
Hasil Penambahan: 5 + 2 = 7
Hasil Pengurangan: 5 - 2 = 3
Hasil Perkalian: 5 x 2 = 10
Hasil Pembagian: 5 / 2 = 2
```

Menambahkan project Windows Form

Setelah menyelesaikan project yang pertama (project CalculatorConsoleApp), latihan berikutnya adalah menambahkan project baru berbasis Windows Form. Untuk langkah-langkahnya seperti berikut:

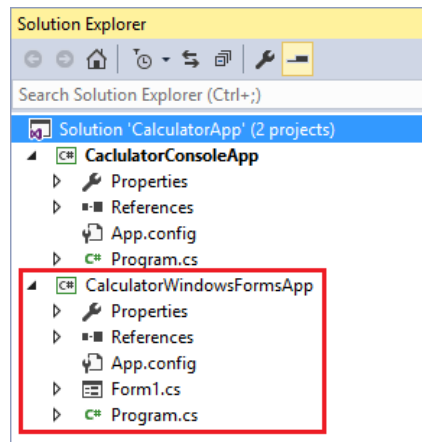
- Masih di solution yang sama, untuk menambahkan project yang kedua dengan cara klik kanan nama *Solution* -> *Add* -> *New Project*.

Kemudian pilih Windows Forms Application, untuk isian Name dan Location menyesuaikan.

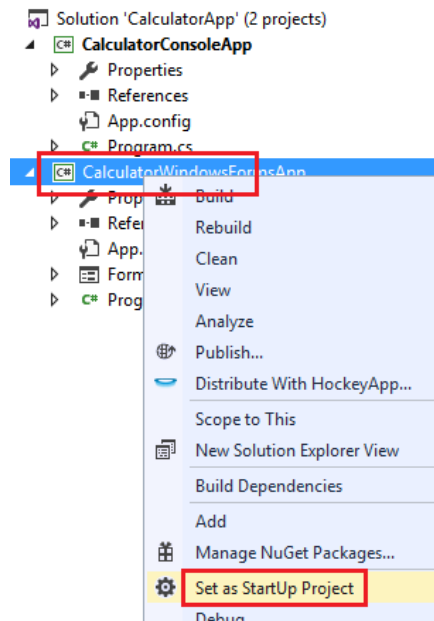


Untuk isian Name diisi dengan **CalculatorWindowsFormsApp**. Penamaan project perlu diperhatikan huruf besar kecilnya karena nama project secara default akan menjadi *namespace* yang akan berpengaruh terhadap penulisan kode program.

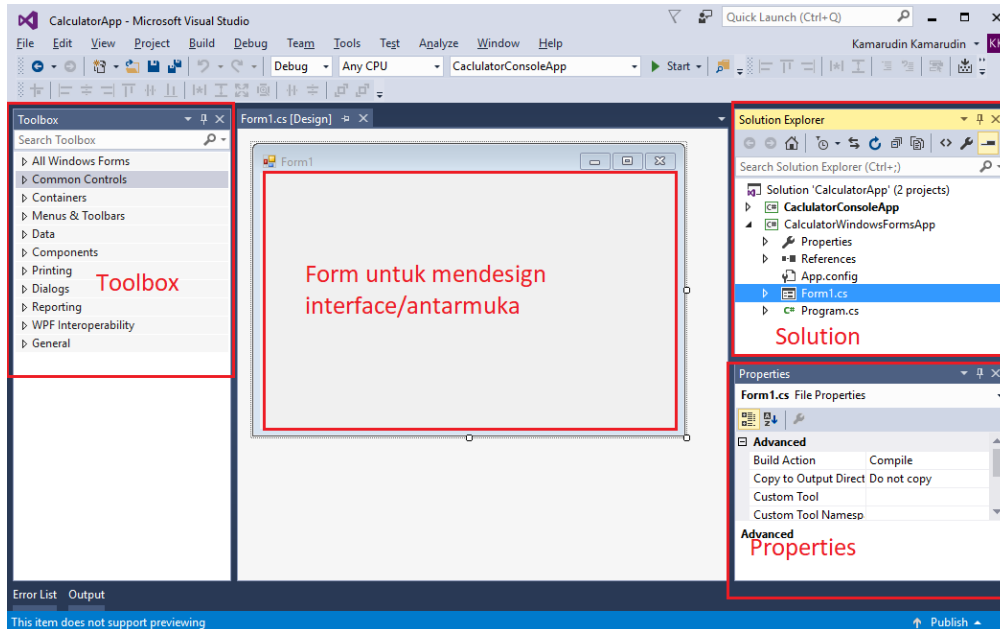
Setelah itu cek di panel solution, seharusnya sudah ada project CalculatorWindowsFormsApp yang baru kita tambahkan.



Karena di latihan ini kita sudah bekerja dengan multi project, langkah berikutnya adalah mengganti pengaturan *Startup Project* agar project yang dijalankan ketika kita menekan tombol F5 adalah project CalculatorWindowsFormsApp. Caranya dengan mengklik kanan project CalculatorWindowsFormsApp -> *Set as Startup Project*.

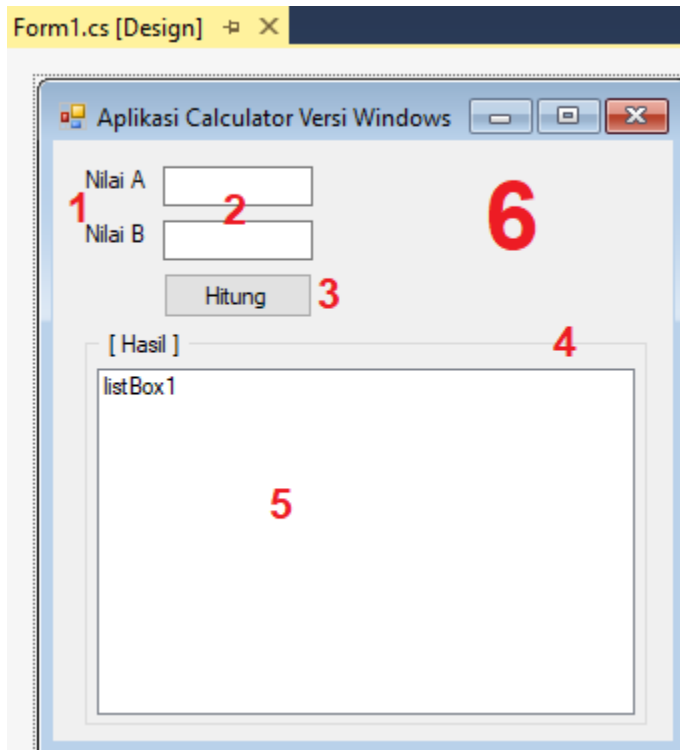


2. Kemudian aktifkan Form1, untuk menampilkan desain windows GUI (Graphical User Interface) seperti berikut:



Pada gambar di atas, panel Toolbox berisi semua komponen yang diperlukan untuk mendesain tampilan aplikasi.

3. Setelah itu kita desain tampilan Aplikasi Calculator seperti gambar berikut:

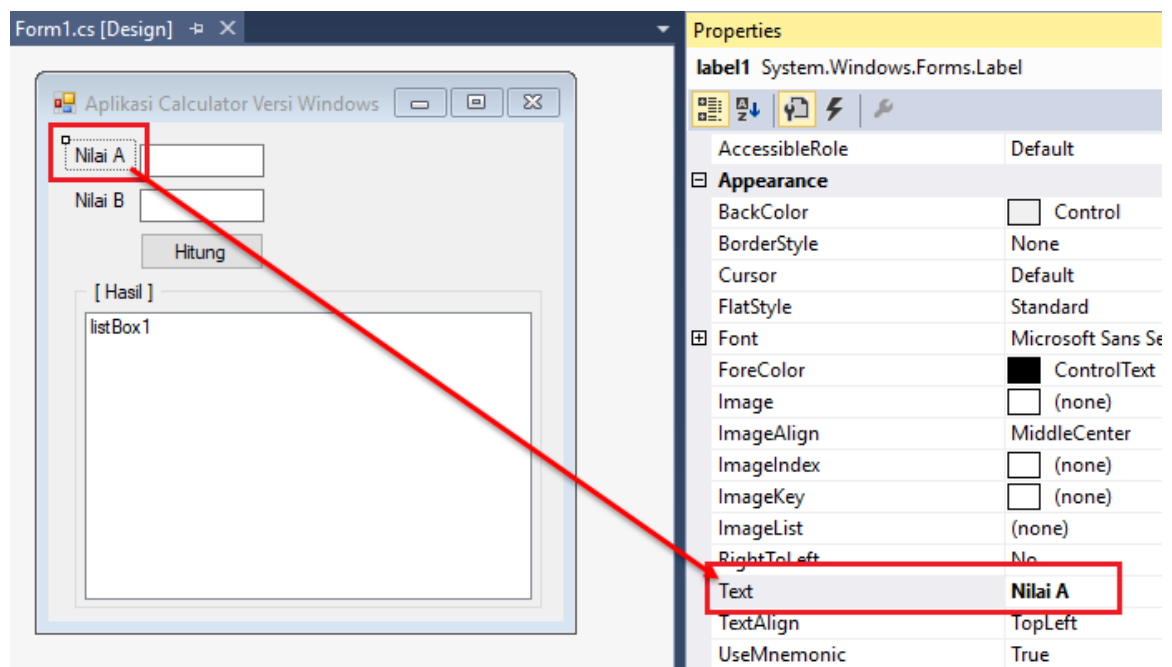


Keterangan gambar

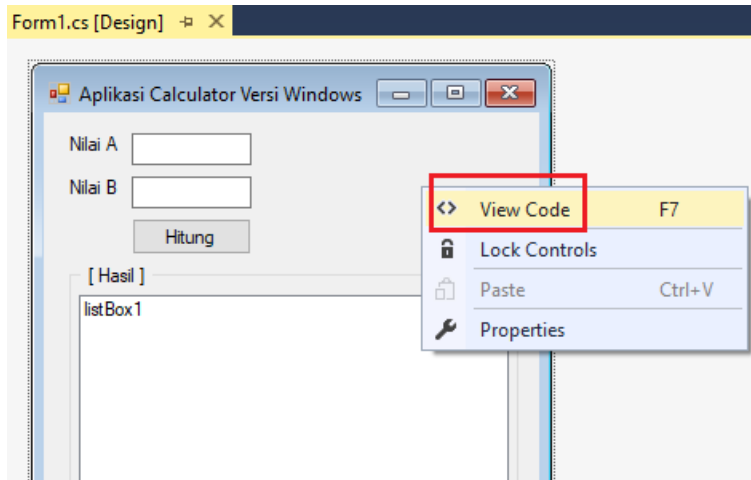
No	Komponen	Property	Nilai/Value
1	Label1 Label2	Text Text	Nilai A Nilai B
2	TextBox1 TextBox2	Name Name	txtNilaiA txtNilaiB
3	Button	Name	btnHitung
4	GroupBox	Text	[Hasil]
5	ListBox	Name	lstHasil
6	Form	Text	Aplikasi Calculator Versi Windows

Khusus untuk komponen *TextBox*, *Button* dan *ListBox* karena nilai property Namanya akan menjadi nama objek di dalam kode program, jadi harus diperhatikan penulisan huruf besar dan kecilnya.

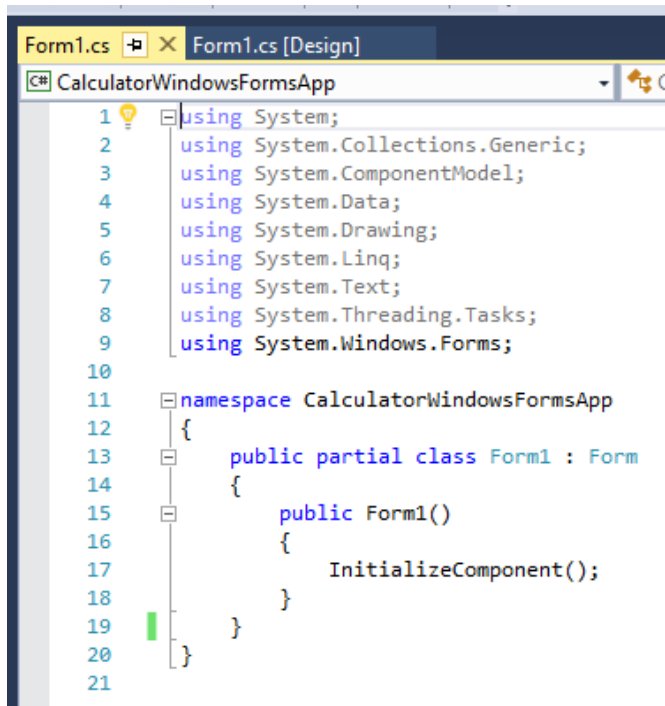
Berikut adalah contoh pengaturan property untuk masing-masing komponen di atas.



- Setelah menyelesaikan desain tampilan Aplikasi Calculator, selanjutnya kita akan menambahkan beberapa method dasar untuk operasi Calculatornya, dengan cara klik kanan form kemudian pilih View Code atau tekan tombol F7.



Setelah itu tampilan akan pindah ke editor code.



Kemudian tambahkan beberapa method/operasi dasar Calculator seperti berikut:

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

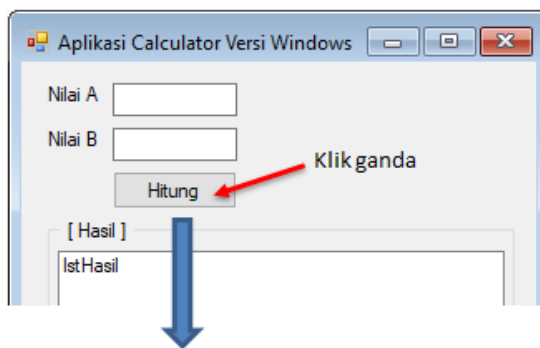
    private int Penambahan(int a, int b)
    {
        return a + b;
    }

    private int Pengurangan(int a, int b)
    {
        return a - b;
    }

    private int Perkalian(int a, int b)
    {
        return a * b;
    }

    private int Pembagian(int a, int b)
    {
        return a / b;
    }
}
```

5. Setelah melengkapi kode di atas, langkah berikutnya adalah kita akan melengkapi kode untuk proses hitung. Caranya adalah dengan mengklik ganda tombol Hitung dan secara otomatis Visual Studio .NET akan membuatkan sebuah method yang bertugas untuk merespon aksi (event) dari user ketika user mengklik tombol Hitung.



```
private void btnHitung_Click(object sender, EventArgs e)
{
}

// Event Handler
```

Method yang bertugas untuk merespon aksi (event) dari user ini disebut dengan method *event handler*. Penamaan method event handler secara default menggunakan format: **namaKomponen_Event**, misal **btnHitung_Click**.

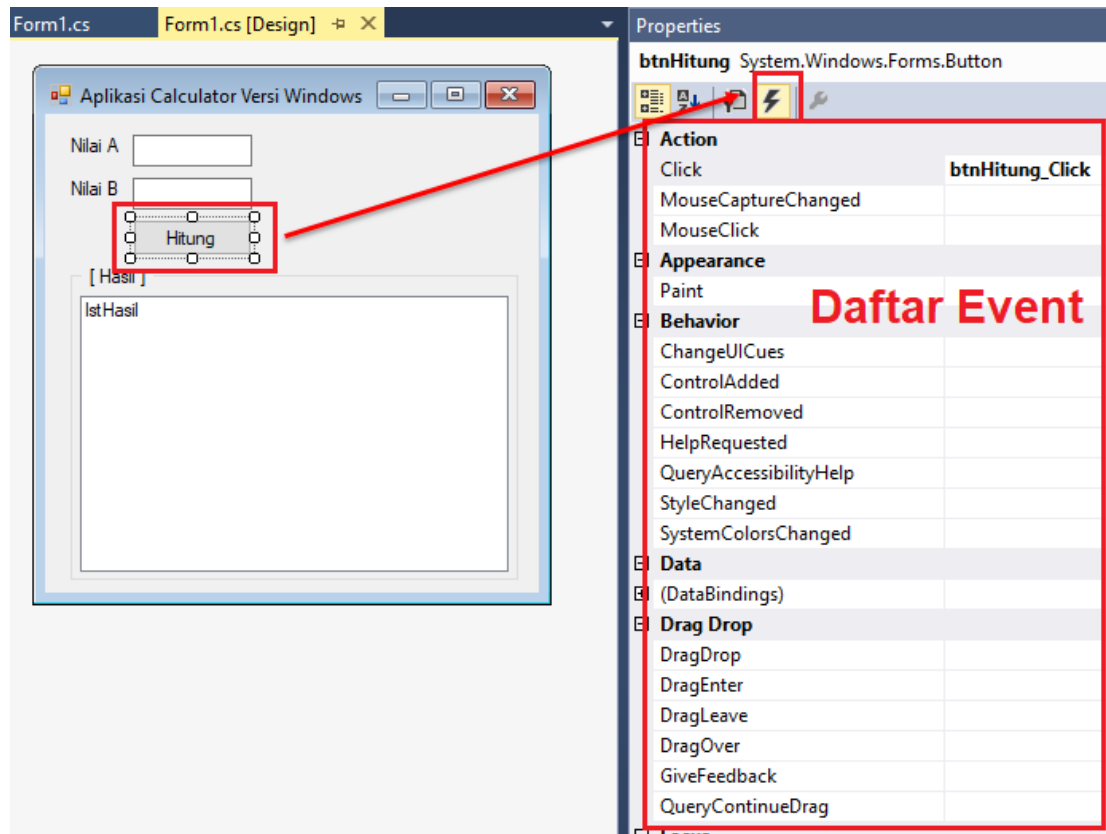
Kemudian lengkapi kode untuk method `btnHitung_Click` seperti berikut:

```
private void btnHitung_Click(object sender, EventArgs e)
{
    var a = int.Parse(txtNilaiA.Text); // casting
    var b = Convert.ToInt32(txtNilaiB.Text); // casting

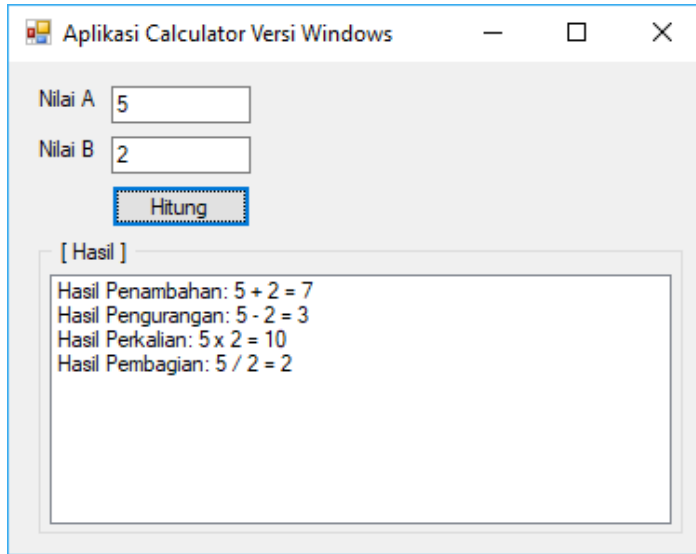
    lstHasil.Items.Clear(); // kosongkan isi listbox

    // tampilkan hasil semua hasil pemanggilan method calculator ke dalam listbox
    lstHasil.Items.Add(string.Format("Hasil Penambahan: {0} + {1} = {2}", a, b, Penambahan(a, b)));
    lstHasil.Items.Add(string.Format("Hasil Pengurangan: {0} - {1} = {2}", a, b, Pengurangan(a, b)));
    lstHasil.Items.Add(string.Format("Hasil Perkalian: {0} x {1} = {2}", a, b, Perkalian(a, b)));
    lstHasil.Items.Add(string.Format("Hasil Pembagian: {0} / {1} = {2}", a, b, Pembagian(a, b)));
}
```

Jika Anda ingin mengetahui event apa saja yang dimiliki oleh sebuah komponen adalah dengan cara memilih komponen tersebut kemudian mengaktifkan panel Properties setelah itu pilih toolbar Events.



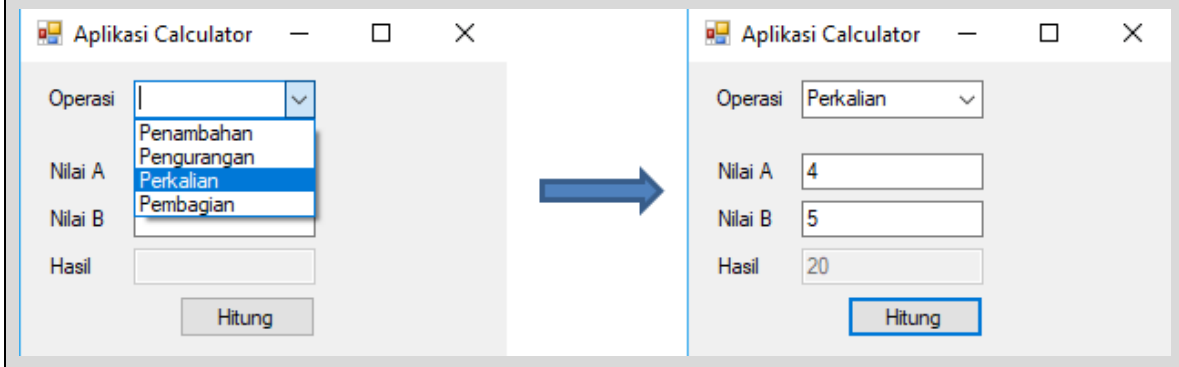
6. Terakhir tekan tombol F5 (Start Debugging) untuk menjalankan program



Tantangan

Buat ulang aplikasi Calculator Windows Form dengan merubah tampilannya seperti gambar di bawah ini.

Kemudian upload hasil source codenya ke account github Anda.



Selesai 😊

Kamarudin, M.Kom
<http://coding4ever.net/>
<https://github.com/rudi-krsoftware/open-retail>