

# VISUAL PROGRAMMING



## PEMROGRAMAN VISUAL

---

Windows Form & Control



# Visual Studio

A MICROSOFT PRODUCT

**TUJUAN PERKULIAHAN**

# TUJUAN

---

- prinsip-prinsip desain antarmuka(GUI).
- Layout WinForm
- Event pada GUI. dan penanganan event.
- Control pada Windows Form



Visual Studio

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

**GUI –EVENT DRIVENT**

# PENDAHULUAN

---

- Graphical User Interface (GUI)
  - Memberikan "**tampilan**" yang khas pada program
  - Dibangun dari control GUI
    - Objek yang dapat menampilkan informasi pada layar atau memungkinkan pengguna untuk berinteraksi dengan aplikasi
    - Mengimplementasikan komponen antarmuka

# GUI

---

- Antarmuka (*user interface*) yang konsisten memungkinkan pengguna untuk mempelajari aplikasi baru lebih cepat.

# EVENT HANDLING

---

- Event Handling
  - GUI adalah *event driven*
  - Ketika pengguna berinteraksi dengan komponen GUI, interaksi dikenal sebagai **event**
  - Sebuah metode yang melakukan tugas dalam menanggapi suatu event disebut **event handler**

# CONTOH EVENT-DRIVEN GUI

---

- Event handler Click pada Button control pada Form
- Setiap event handler menerima dua parameter ketika dipanggil
  - object named *sender*
    - Sebuah referensi object yang mengenerated event
  - EventArgs named *e*
    - Mengandung informasi tambahan tentang peristiwa yang terjadi



```

1 // contoh event
2 // kuliah pertemuan 2 event handlers.
3 using System;
4 using System.Windows.Forms;
5
6 // Form contoh sederhana event handler
7 public partial class Form1 : Form
8 {
9     // default constructor
10    public Form1()
11    {
12        InitializeComponent();
13    } // end constructor
14
15    // handles click event of Button clickButton
16    private void Button1_Click( object sender, EventArgs e )
17    {
18        MessageBox.Show( "Klik Tombol." );
19    } // end method Button1_Click
20 } // end class Form

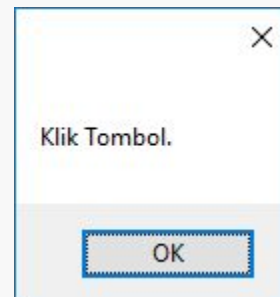
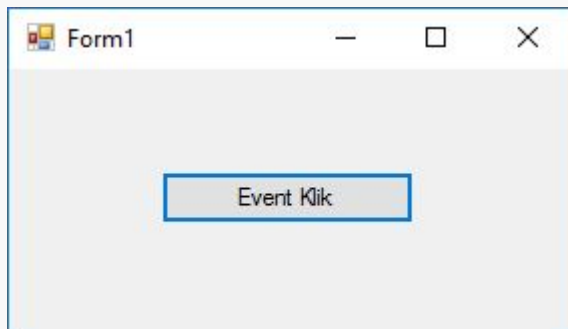
```

Inheritance dari **Form**

Visual Studio generated  
**InitializeComponent**

event handler klik untuk **Button1**

Aksi dari event klik menampilkan  
MessageBox



# VISUAL STUDIO MEN-GENERATE CODE

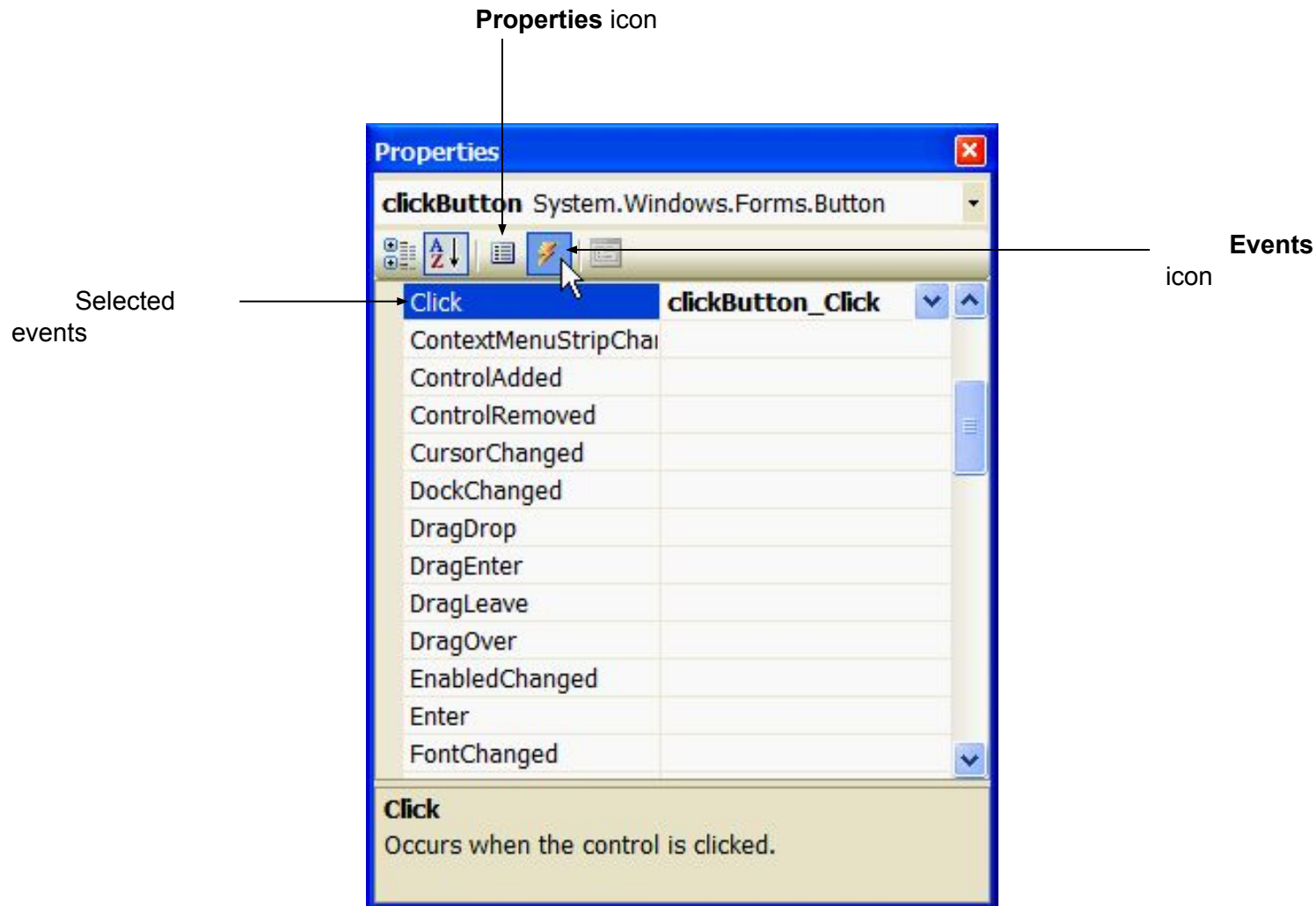
- Visual Studio men-generated Code
  - Code yang di generate pada file Designer.cs pada Form
  - Code yang degenerate dengan partial
  - Secara default, semua variable memiliki private access modifier
  - Code juga memiliki Dispose dan InitializeComponent

# ERROR-PREVENTION

---

- ❑ Kode yang dihasilkan dengan membangun GUI dalam mode Desain tidak dimaksudkan untuk dimodifikasi langsung, modifikasi dapat mengakibatkan sebuah aplikasi tidak berfungsi (error).
- ❑ Anda harus memodifikasi melalui jendela Properties.

# MELIHAT EVENT UNTUK TOMBOL KONTROL PADA JENDELA PROPERTIES.





Visual Studio

A Microsoft Product

**FORM**

# WINDOWS FORMS

---

- Windows Forms
  - Digunakan untuk membuat GUI untuk program
  - Elemen grafis yang muncul di desktop komputer Anda
  - Form merupakan **container** untuk kontrol dan komponen
  - Dalam pemrograman visual, Visual Studio menghasilkan/mengenerate kode yang berhubungan dengan GUI

# FORM PROPERTIES

---

- properti FORM biasanya mengontrol tampilan visual:
  - AutoScroll
  - BackgroundImage
  - ControlBox
  - FormBorderStyle (sizable?)
  - Icon
  - Location
  - Size
  - StartPosition
  - Text (i.e. window's caption)
  - WindowState (minimized, maximized, normal)

# FORM METHODS

---

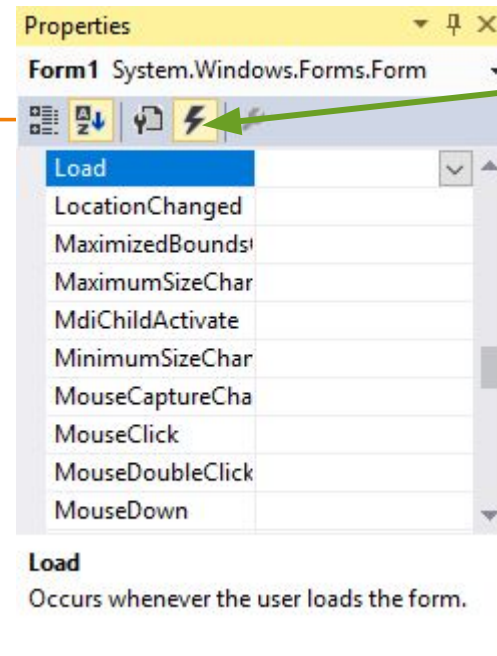
## □ Aksi yang bisa dilakukan pada Form:

```
form.Hide();  
.  
.  
.  
form.Show();
```

- Activate : *give this form the focus*
- Close : *close & release associated resources*
- Hide : *hide, but retain resources to show form later*
- Refresh : *redraw*
- Show : *make form visible on the screen, & activate*
- ShowDialog: *show modally*



# FORM EVENTS



- Load : *occurs just before form is shown for first time*
- Closing : *occurs as form is being closed (ability to cancel)*
- Closed : *occurs as form is definitely being closed*
- Resize : *occurs after user resizes form*
- Click : *occurs when user clicks on form's background*
- KeyPress : *occurs when form has focus & user presses key*



Visual Studio

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

**CONTAINER**

# CONTAINER

---

- ❑ *Container controls form a sub-category within controls i.e. some controls are containers.*
- ❑ *A container is basically something that can contain other controls.*
- ❑ *Every container has a configurable layout which defines how the child controls of a container are positioned relative to each other – simple examples are horizontally and vertically.*
- ❑ *This ability to have a layout is the most important aspect of a container.*

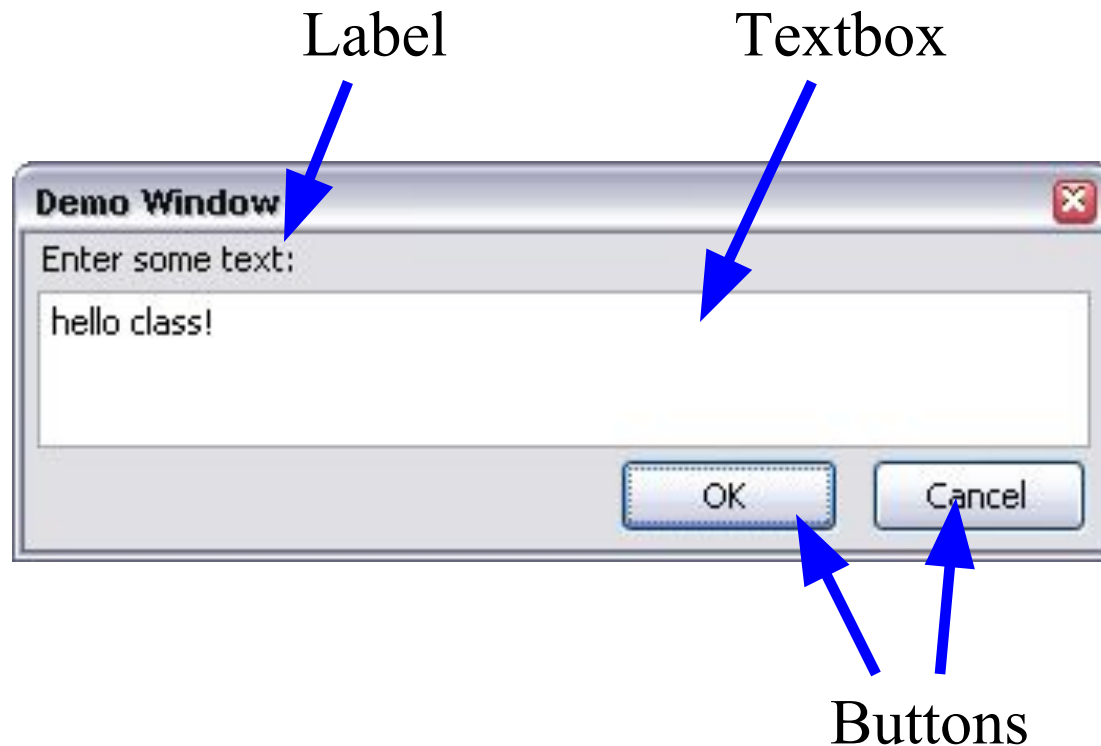
# CONTAINER

---

- Komponen yang mengandung satu atau lebih komponen lain
- Membuat struktur antarmuka pengguna
- Mengelola komponen anak
  - Layout, pengiriman event, dll
- Beberapa memiliki fitur interaktif (mis. Panel tab)

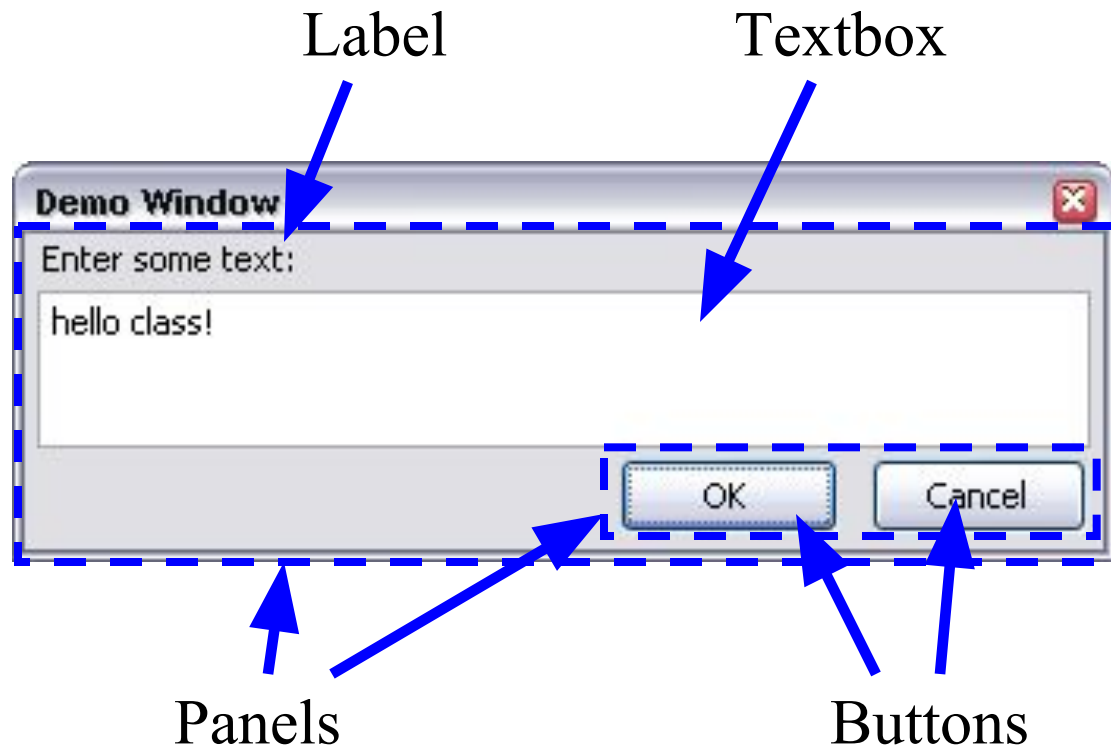
# STRUKTUR CONTAINER

---

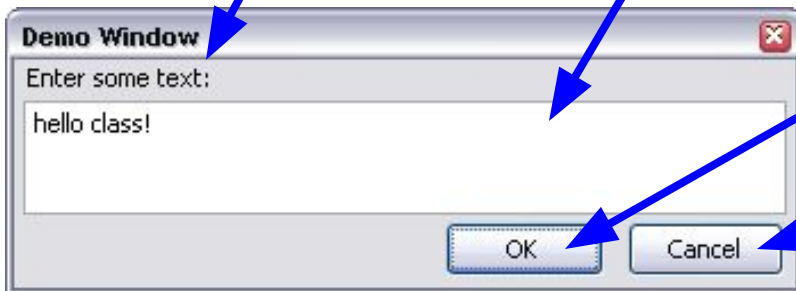
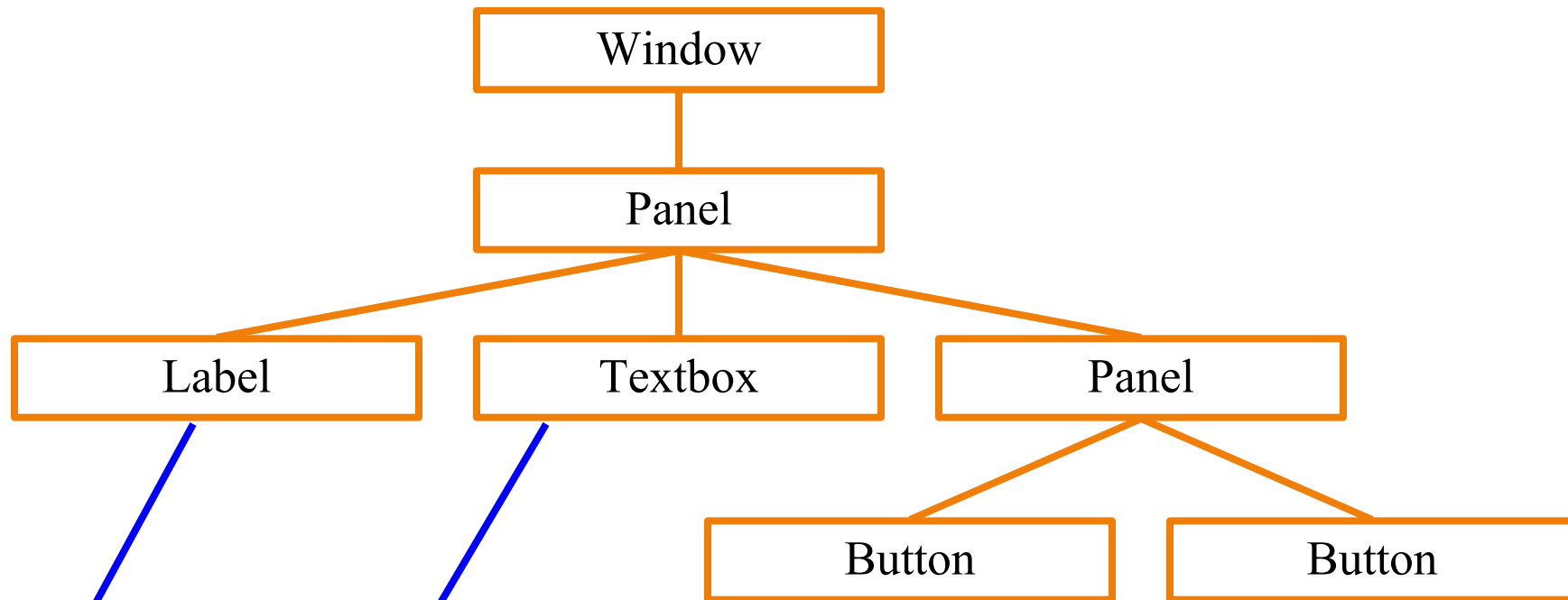


# STRUKTUR CONTAINER

---



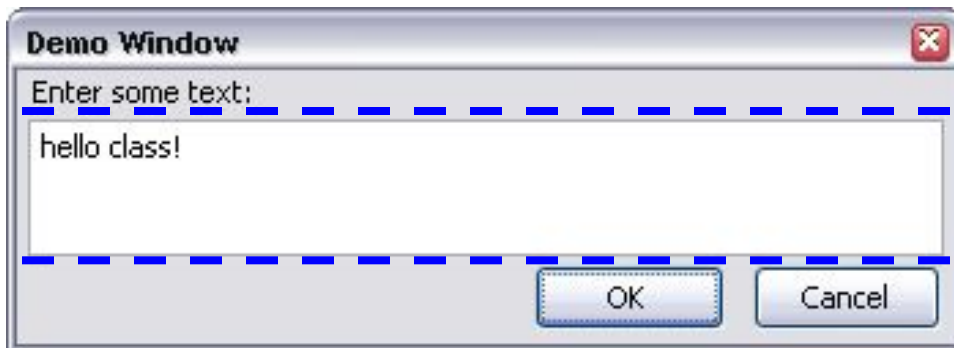
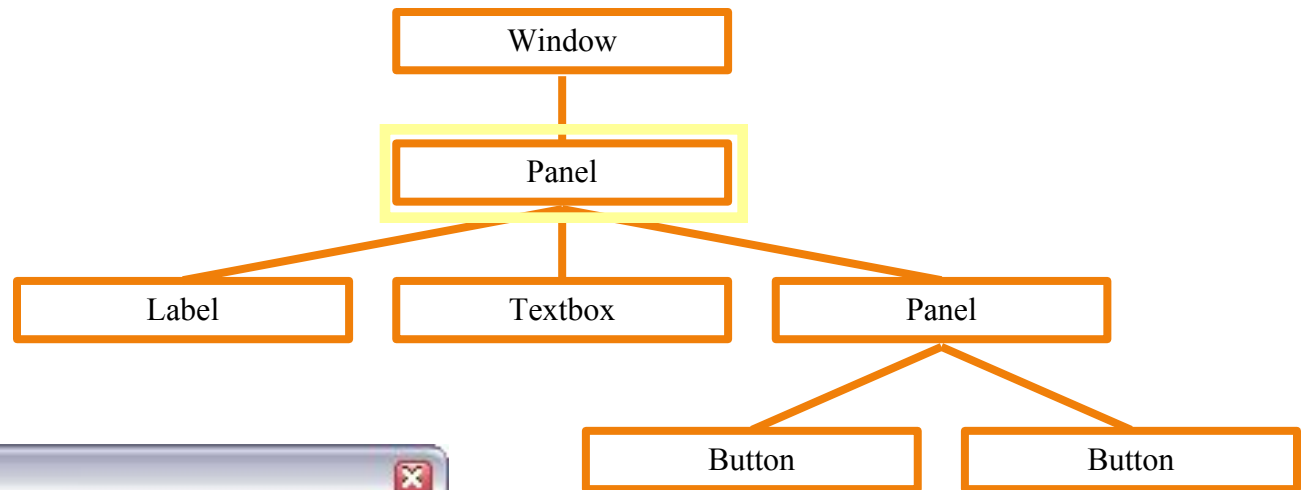
# STRUKTUR CONTAINER



# LAYOUT

---

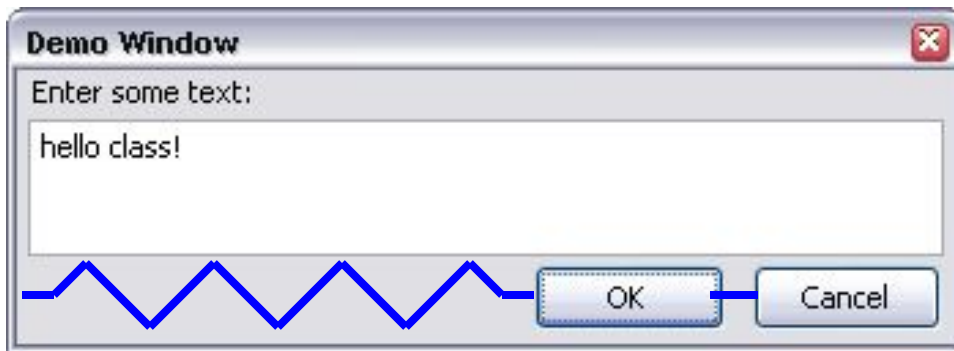
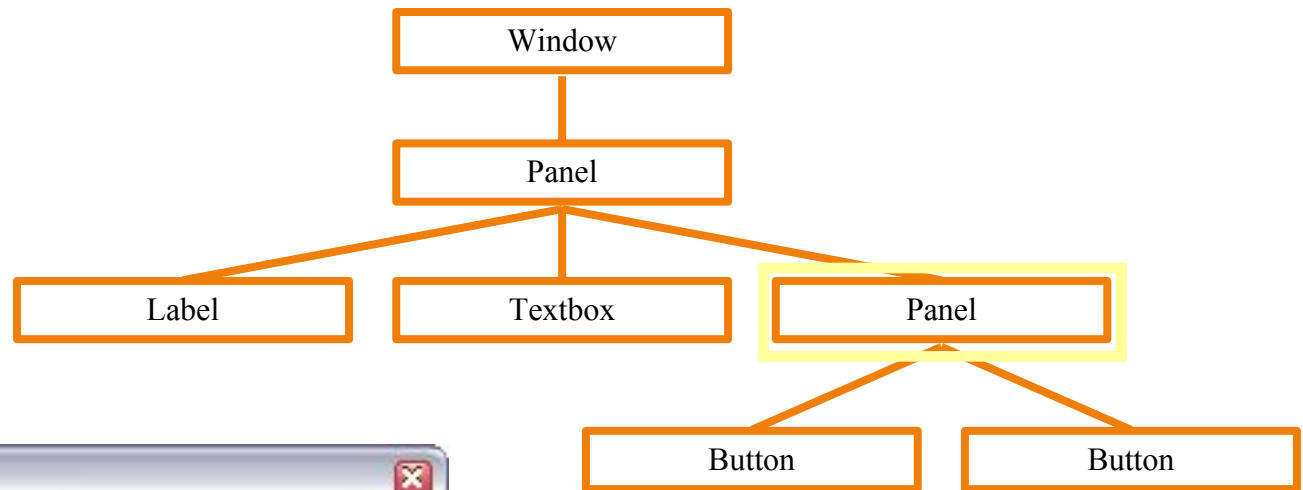
Containers specify layout of their children





# LAYOUT

## □ Mengelola komponen anak



spring

strut



Visual Studio

A MICROSOFT PRODUCT

**CONTAINER**

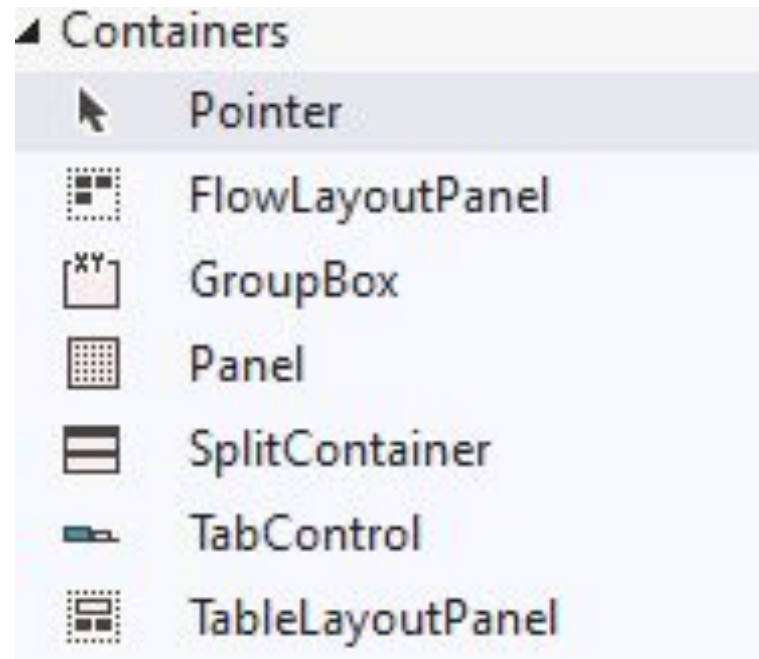
# LAYOUT FORM

---

- Seperti yg dibahas pada subsection sebelumnya bahwa container juga mengatur tata letak komponen control yang ada didalamnya

# LAYOUT PADA .NET

---



# CONTOH : FLOW LAYOUT

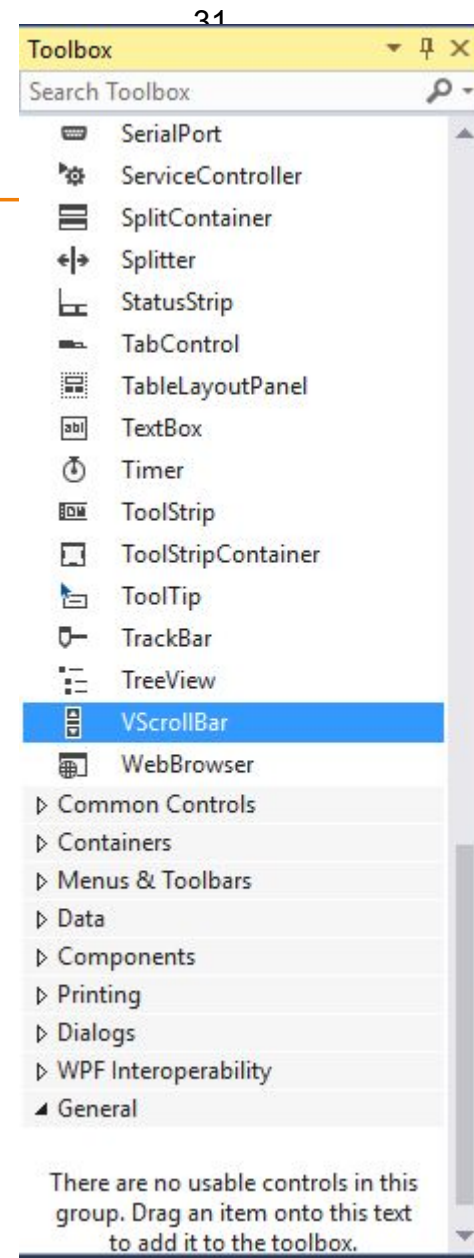
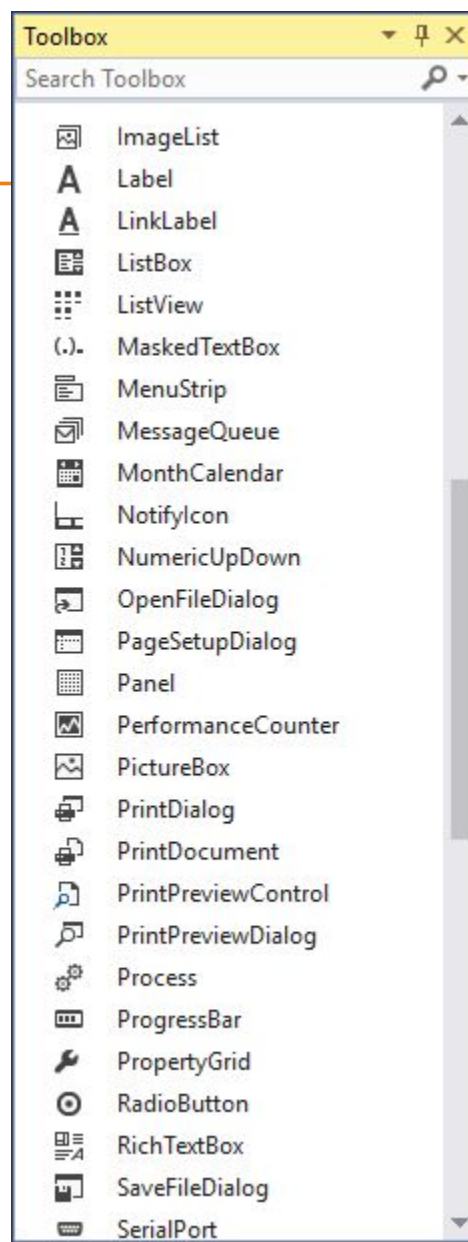
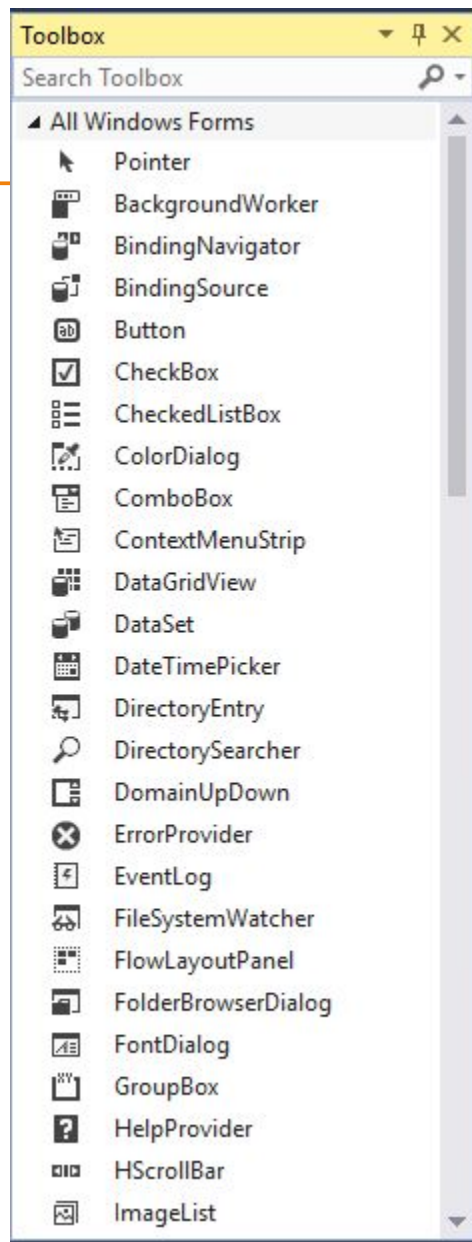
---



Visual Studio

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

**KOMPONEN CONTROL**



# CONTROL DAN KOMPONEN

- 
- <https://docs.microsoft.com/en-us/dotnet/api/system.windows.forms.control?view=netframework-4.8>



# YANG PERLU DILAKUKAN

---

- Mengetahui penggunaan komponen
- Masing-masing komponen berbasis *class* (*oop class*)
- *Konsep OOP Kelas memiliki*
  - *Atribut/property*
  - *method*

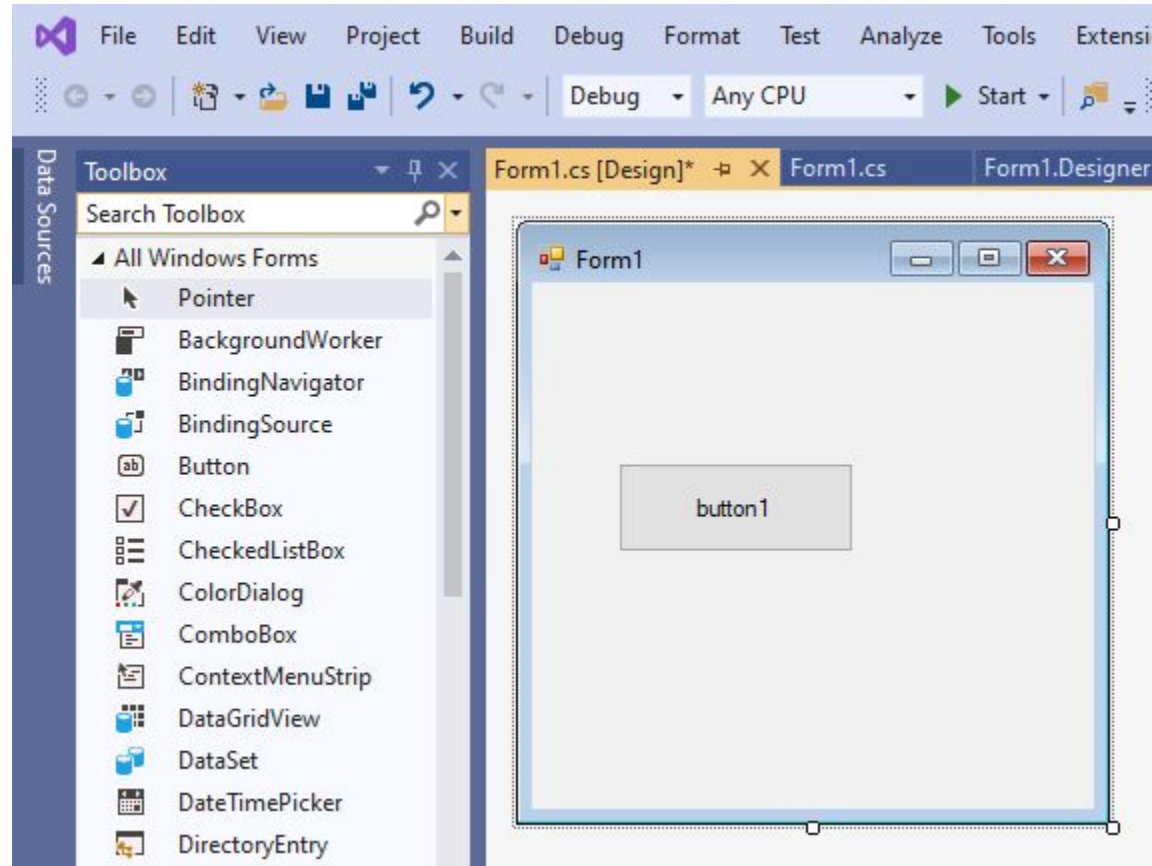
---

# “WELL-KNOWN” EVENT

*Secara teknis program  
event terkait method  
pada masing-masing  
komponen*

# CONTOH

## □ Button



# CODE YG DEGENERATE VS

---

```
private void InitializeComponent()
{
    this.button1 = new System.Windows.Forms.Button();
    this.SuspendLayout();
    //
    // button1
    //
    this.button1.Location = new System.Drawing.Point(43, 90);
    this.button1.Name = "button1";
    this.button1.Size = new System.Drawing.Size(118, 45);
    this.button1.TabIndex = 0;
    this.button1.Text = "button1";
    this.button1.UseVisualStyleBackColor = true;
    //
    ..
}
```

# PROPERTY

PROPERTI	DESKRIPSI
BackColor	Menggunakan properti BackColor Anda dapat mengatur warna latar belakang tombol.
BackgroundImage	Menggunakan popor BackgroundImage Anda dapat mengatur gambar latar belakang pada tombol.
AutoEllipsis	Menggunakan properti AutoEllipsis Anda dapat menetapkan nilai yang menunjukkan bahwa apakah karakter ellipsis (...) muncul di tepi kanan kontrol yang menunjukkan bahwa teks tombol melampaui melampaui panjang tombol yang ditentukan.
<u>AutoSize</u>	Menggunakan properti AutoSize Anda dapat menetapkan nilai yang menunjukkan apakah tombol mengubah ukuran berdasarkan isinya.
Enable	Menggunakan properti enable/Diaktifkan Anda dapat menetapkan nilai yang menunjukkan apakah tombol dapat merespons interaksi pengguna.
Event	Menggunakan properti event Anda bisa mendapatkan daftar penanganan acara yang diterapkan pada tombol yang diberikan.
Font	Menggunakan properti Font, Anda dapat mengatur font tombol.
FontHeight	Menggunakan properti FontHeight Anda dapat mengatur ketinggian font.
FontColor	Menggunakan properti ForeColor Anda dapat mengatur warna latar depan tombol.
Height	Menggunakan properti Height Anda dapat mengatur ketinggian tombol.
Image	Menggunakan properti Image Anda dapat mengatur gambar pada tombol.
Margin	Menggunakan properti Margin Anda dapat mengatur margin antara kontrol.
Name	Menggunakan properti Name, Anda dapat mengatur nama tombol.
Padding	Menggunakan properti Padding Anda dapat mengatur padding di dalam tombol.
<u>Visible</u>	Menggunakan properti Visible Anda dapat menetapkan nilai yang menunjukkan apakah tombol dan semua tombol turunannya ditampilkan.

# EVENT

EVENT	DESKRIPSI
<b>Click</b>	Peristiwa ini terjadi ketika tombol diklik.
<b>DoubleClick</b>	Peristiwa ini terjadi ketika pengguna melakukan klik dua kali pada tombol.
<b>Enter</b>	Peristiwa ini terjadi ketika kontrol dimasukkan.
<b>KeyPress</b>	Peristiwa ini terjadi ketika karakter, atau spasi, atau tombol backspace ditekan saat kontrol memiliki fokus.
<b>Leave</b>	Peristiwa ini terjadi ketika fokus input meninggalkan kontrol.
<b>MouseClick</b>	Peristiwa ini terjadi ketika Anda mengklik pointer mouse pada tombol.
<b>MouseDoubleClick</b>	Peristiwa ini terjadi ketika Anda mengklik dua kali pointer mouse pada tombol.
<b>MouseHover</b>	Peristiwa ini terjadi ketika penunjuk mouse ditempatkan pada tombol.
<b>MouseLeave</b>	Peristiwa ini terjadi ketika pointer mouse meninggalkan tombol.

# MENULIS CODE UNTUK BUTTON

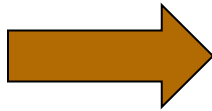
- Setiap kontrol memiliki event default.
- Event default untuk kontrol Button Klik.
- Kita harus menulis kode ke dalam function handler yang akan dijalankan ketika tombol ditekan.
- Untuk membuat fungsi handler, cukup klik dua kali untuk tombol saat Anda berada dalam tampilan desainer.

# MENULIS CODE UNTUK BUTTON

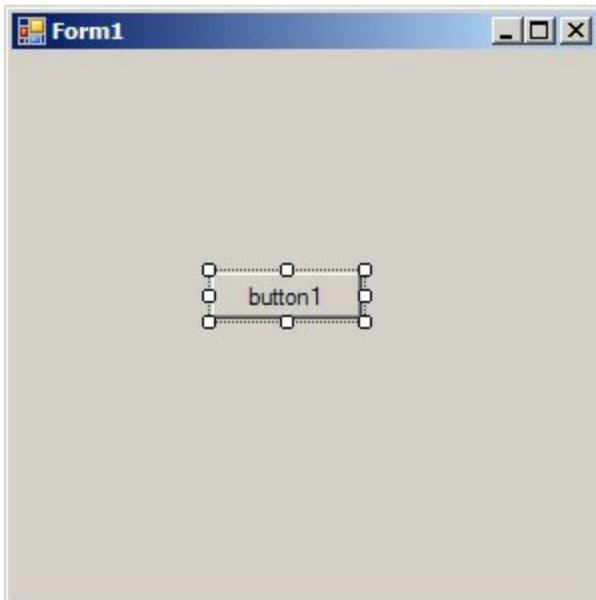
---

Double Click

Designer View



Generated code



```
namespace MyApplication
{
    public partial class frmMain : Form
    {
        public frmMain()
        {
            InitializeComponent();
        }

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



# LAYOUT

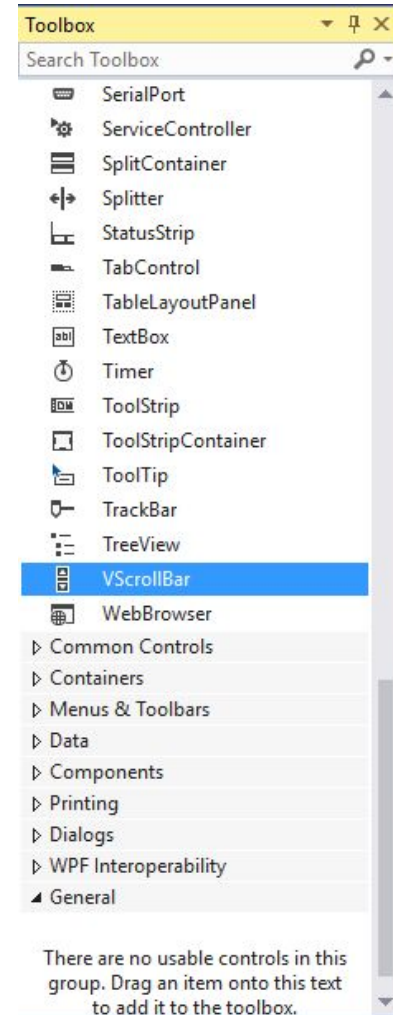
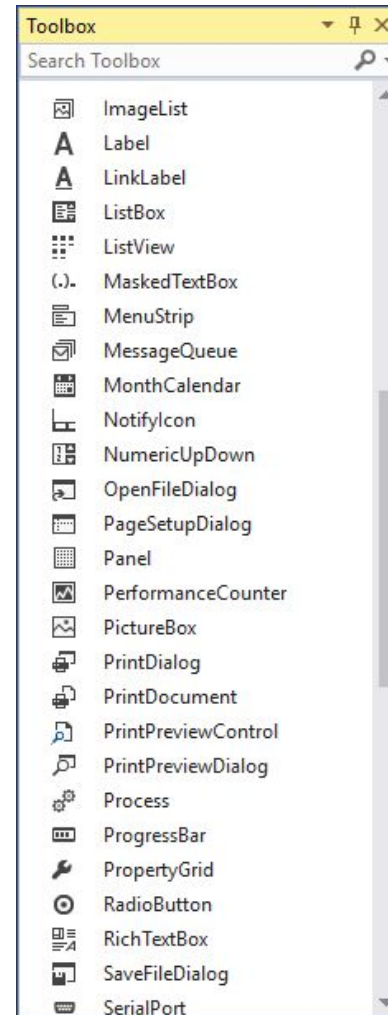
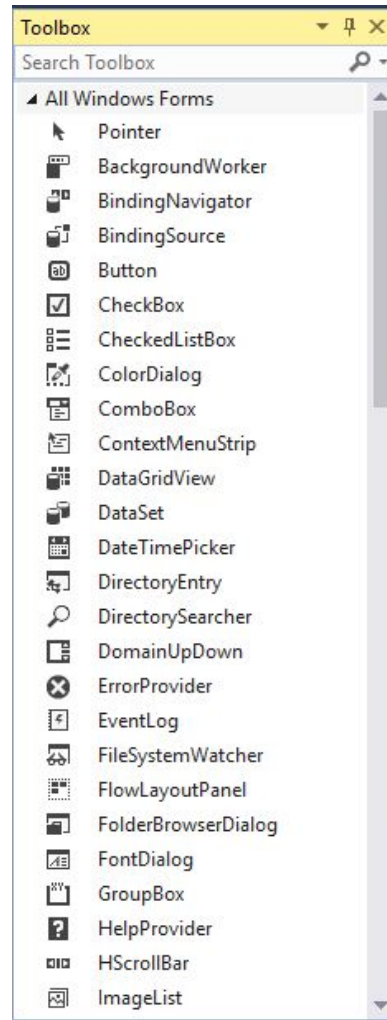
---

## □ Control Properties dan Layout

- Focus method
  - Memindahkan fokus ke kontrol dan membuat kontrol aktif
- Enabled property
  - Menunjukkan apakah pengguna dapat berinteraksi dengan kontrol untuk menghasilkan event
- Anchoring property
  - .Menyebabkan kontrol memiliki jarak tetap dari sisi container
- Docking property
  - Menempel kontrol pada container sehingga control membentang di seluruh sisi
- Padding property
  - Menentukan jarak antara kontrol dan tepi Form
- Width and Height
  - Menentukan ukuran Form

## □ Snap lines

# KOMPONEN LAIN.....



---

Do the same thing as you need.....

Learn by **DOING**.



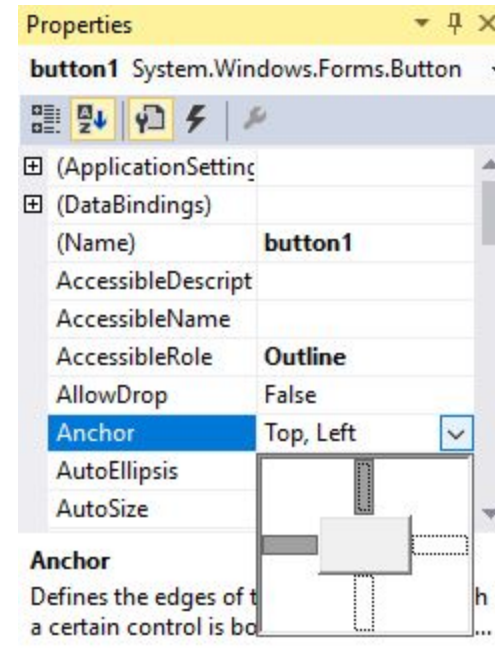
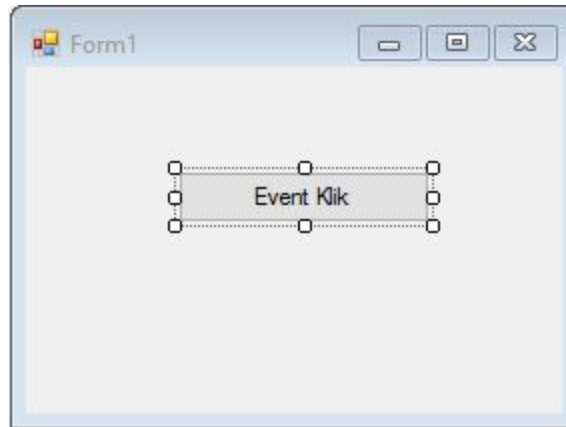


# Visual Studio

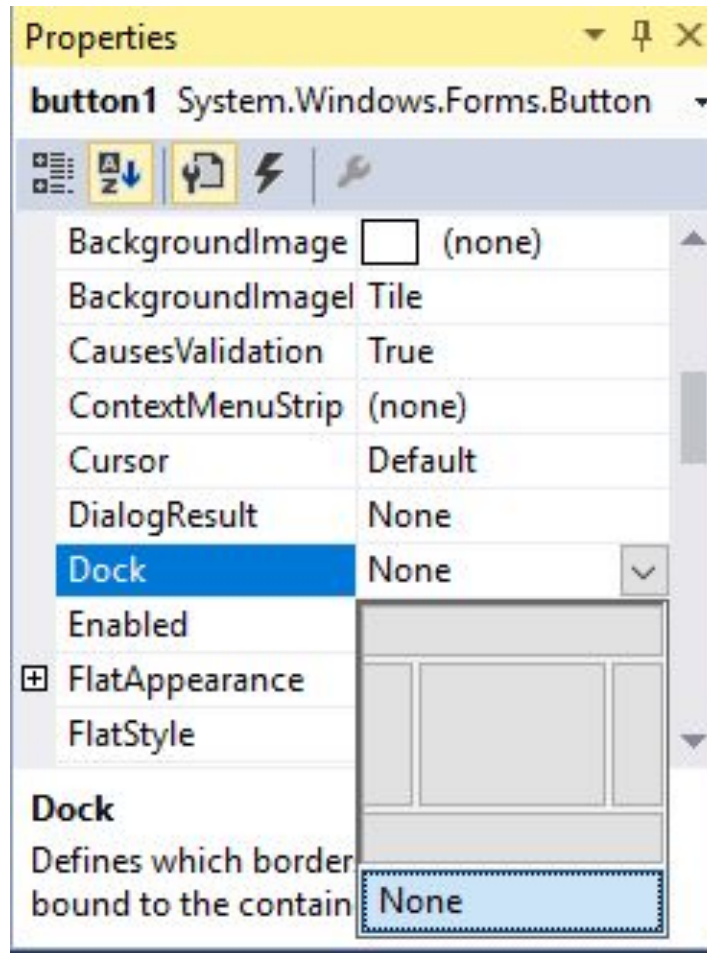
A MICROSOFT PRODUCT

**ANCHOR**

# MANIPULASI ANCHOR



# DOCKING BUTTON PADA FORM.



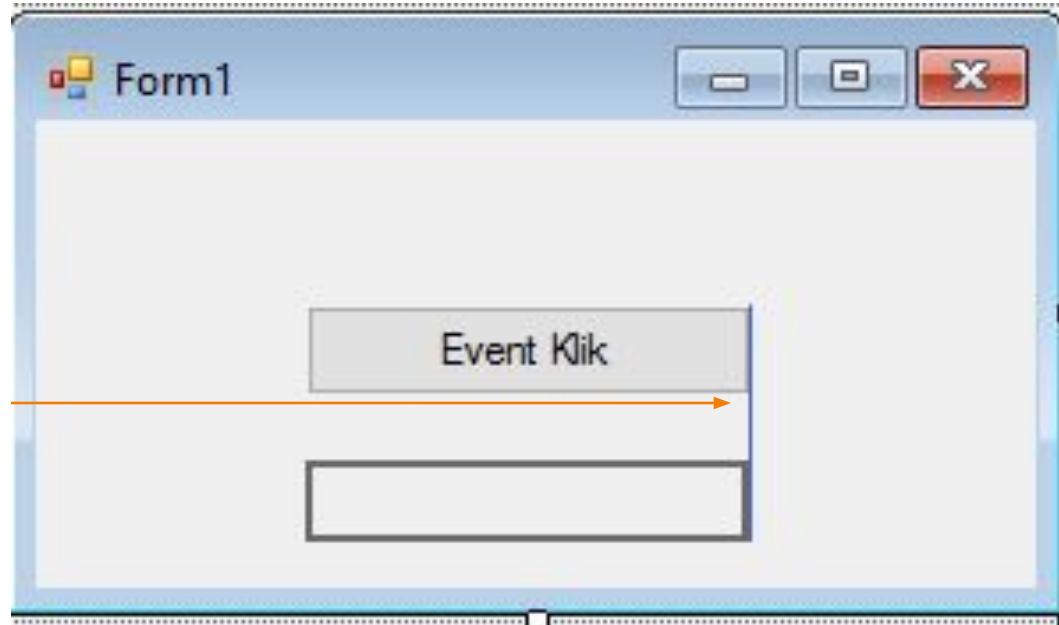
# CONTROL LAYOUT PROPERTIES.

Control layout properties	Description
Anchor	Causes a control to remain at a fixed distance from the side(s) of the container even when the container is resized.
Dock	Allows a control to span one side of its container or to fill the entire container.
Padding	Sets the space between a container's edges and docked controls. The default is 0, causing the control to appear flush with the container's sides.
Location	Specifies the location (as a set of coordinates) of the upper-left corner of the control, in relation to its container.
Size	Specifies the size of the control in pixels as a Size object, which has properties Width and Height.
MinimumSize, MaximumSize	Indicates the minimum and maximum size of a Control, respectively.

# SNAP LINES

---

Snap line to help align controls on their left sides







Menu

Tugas

Kesimpulan

Referensi

Bantuan

Keluar

---

# EXAMPLE



Menu

Tugas

Kesimpula

Referensi

Bantuan

Keluar

# CODE CHALLENGE

---



Menu

Tugas

Kesimpulan

Referensi

Bantuan

Keluar

# QUESTION





Menu

Tugas

Kesimpulan

Referensi

Bantuan

Keluar

# KESIMPULAN

---

CONCLUSIÓN





Menu

Tugas

Kesimpulan

Referensi

Bantuan

Keluar

# REFERENSI

---



Menu

Tugas

Kesimpula

Referensi

Bantuan

Keluar

## Bantuan



Tombol "Home", itu akan membawa Anda kembali ke halaman pembuka.



Tombol kembali ke halaman terakhir yang dilihat

Menu

Tombol menu materi utama dalam media .

Tugas

Tombol tugas untuk menuju ke halaman tugas.

Kesimpulan

Tombol Kesimpulan untuk menuju ke halaman kesimpulan

Referensi

Tombol referensi untuk menuju ke halaman referensi

Bantuan

Tombol bantuan untuk menuju ke halaman bantuan

Keluar

Tombol untuk menutup aplikasi

---

SELESAI