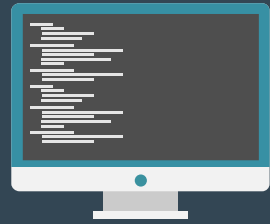
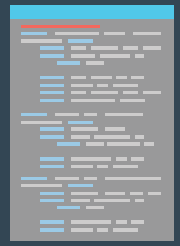


Form

Linear



Build

Event Handling



PROGRAMMING II



Comscience@BRU By Zagon Bussabong

# Chapter 5

Windows Form Application

# Intro



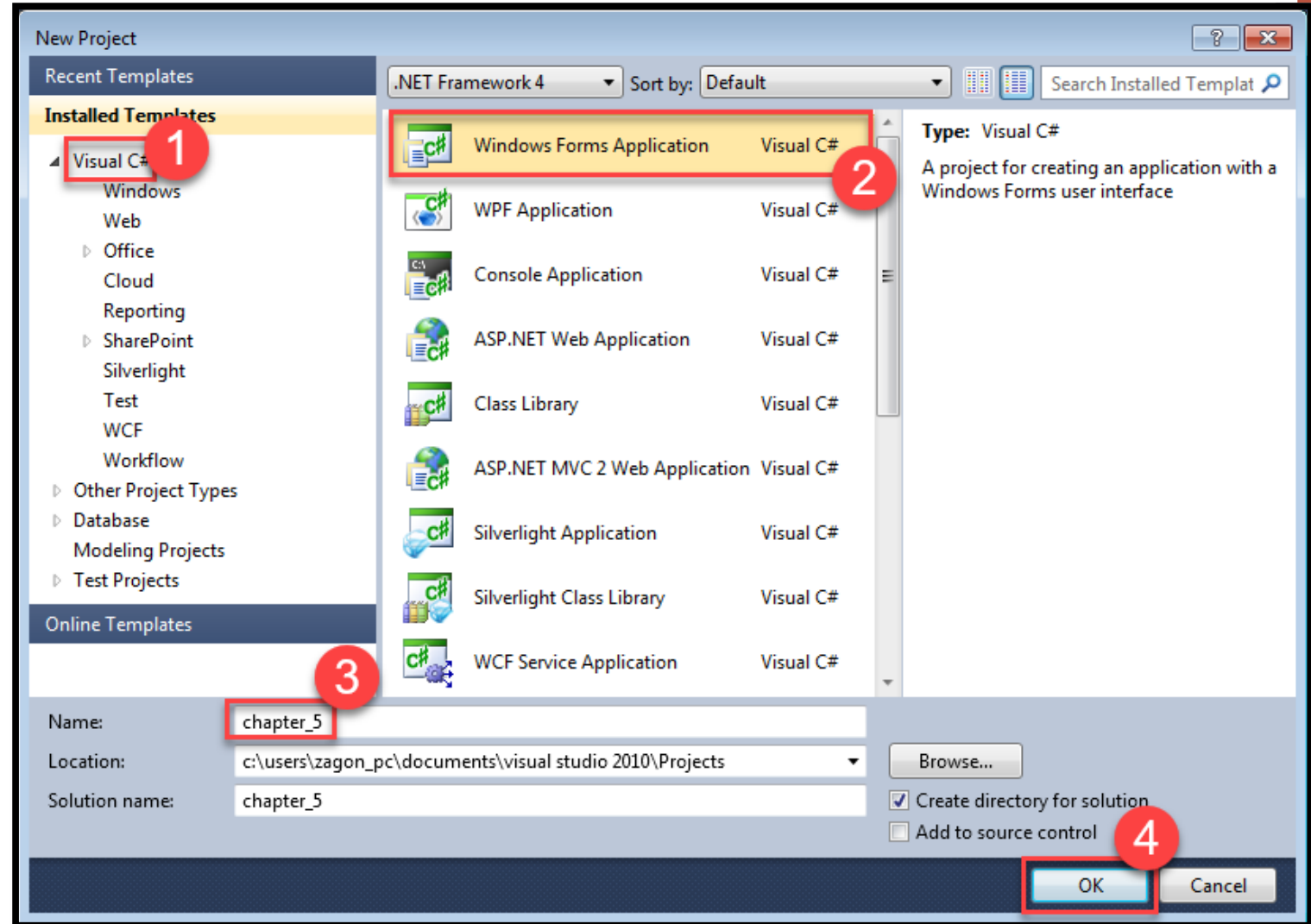
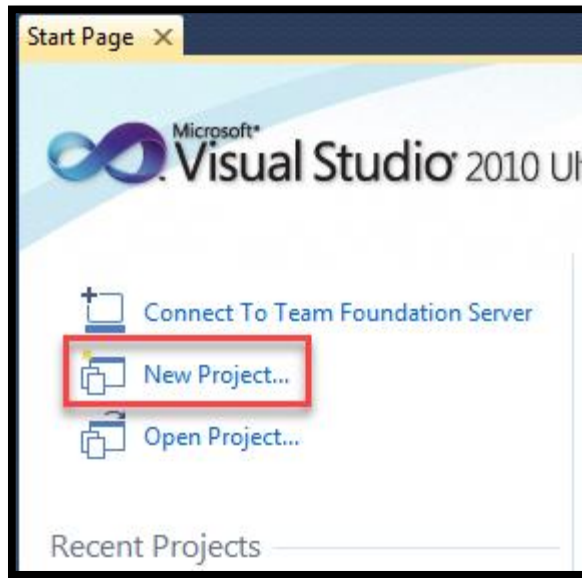
- การเขียนโปรแกรมโดยปกติจะเป็นการเขียนโดยมี GUI ที่ใช้ติดต่อกับผู้ใช้
- สำหรับ Visual Studio C# เราเรียกส่วน GUI ว่า Windows Form ที่เคยลองกันไป  
แล้ว
- ในบทนี้จะมาสร้าง Windows Form Application รวมถึงลองใช้งาน Control ต่างๆ



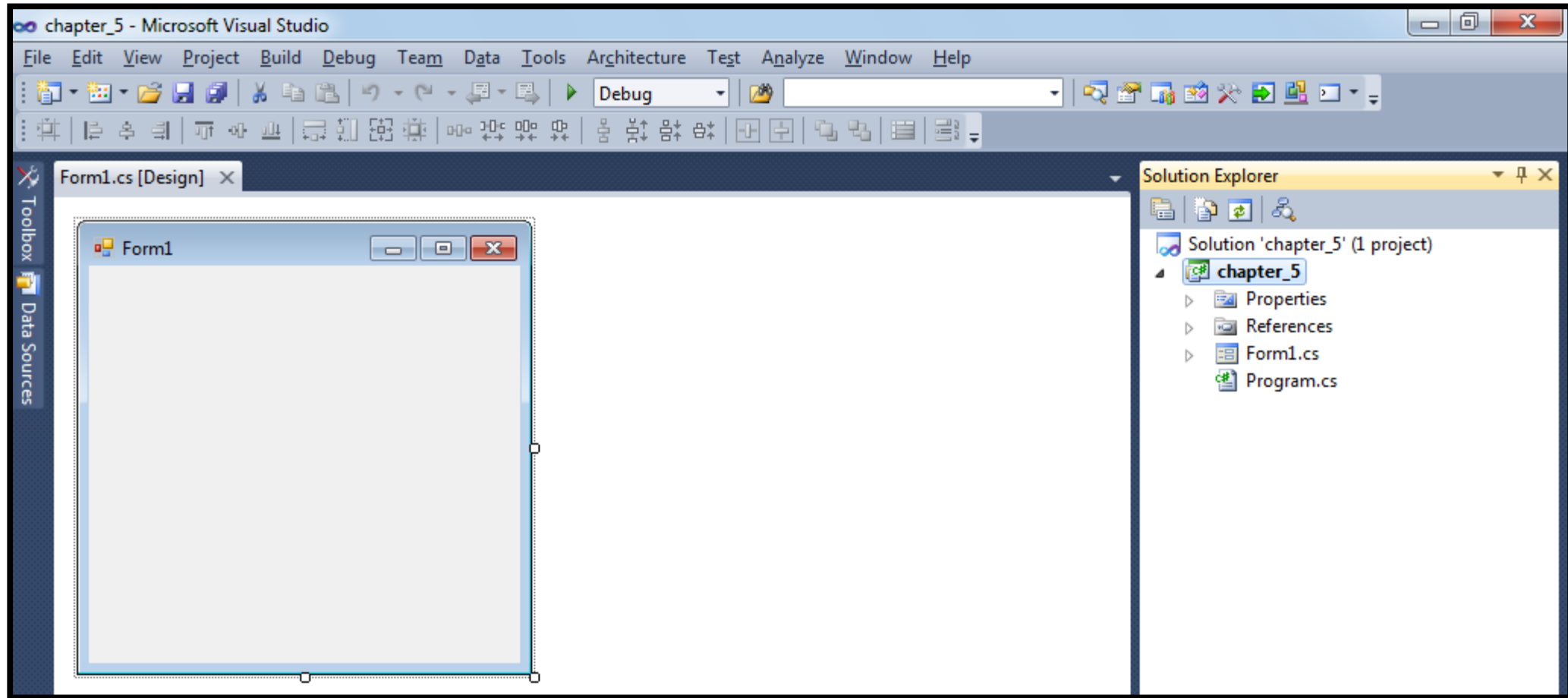
# Structure of Windows Form Application

- การเขียนโปรแกรมแบบ Windows Form Application มีโครงสร้างต่างจาก Console Application
- Windows Form Application เป็นการเขียน code แบบ Object-oriented ล้วนๆ
- การที่จะศึกษา Windows Form Application ได้นั้นจำเป็นต้องเขียนโปรแกรมแบบ Console Application เป็นแล้ว เนื่องจาก Windows Form Application จะมีการสร้างไฟล์ต่างๆมาให้มากมาย จึงมักเกิดความสับสนกับผู้ที่ไม่ชำนาญ

# Structure of Windows Form Application



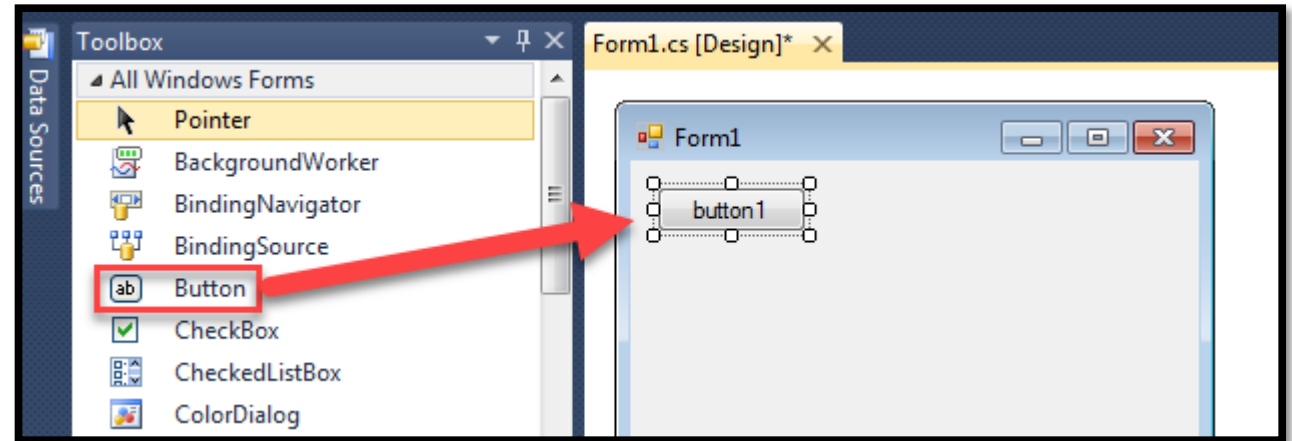
# Structure of Windows Form Application






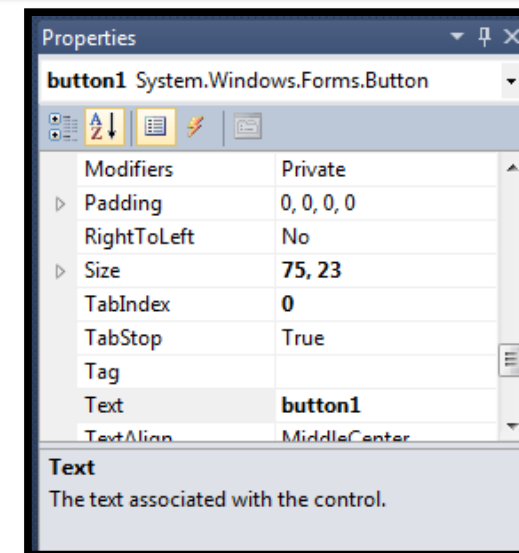
# Structure of Windows Form Application

- นำ Button มาใส่ใน Form1



- จากนั้นดูที่ Properties Windows

- หากไม่พบให้ดูที่  หรือกด Ctrl+w, p

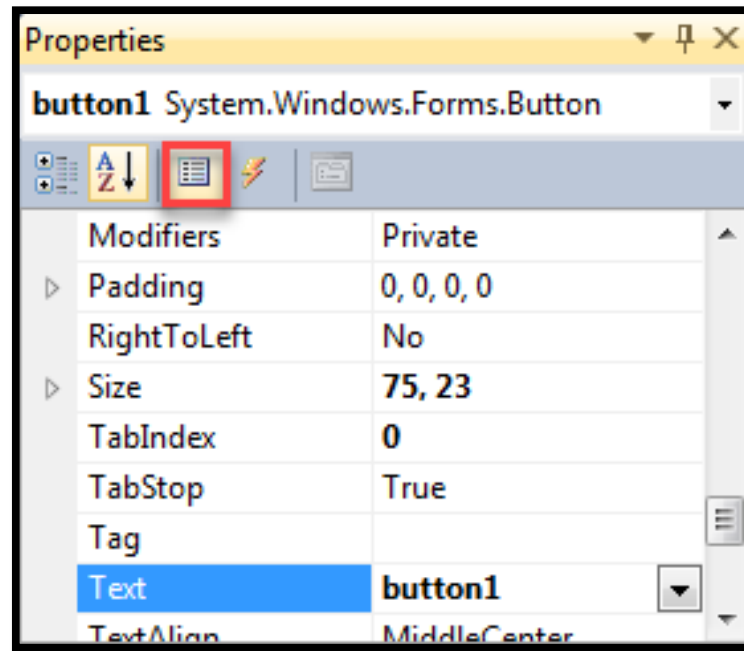


# Structure of Windows Form Application

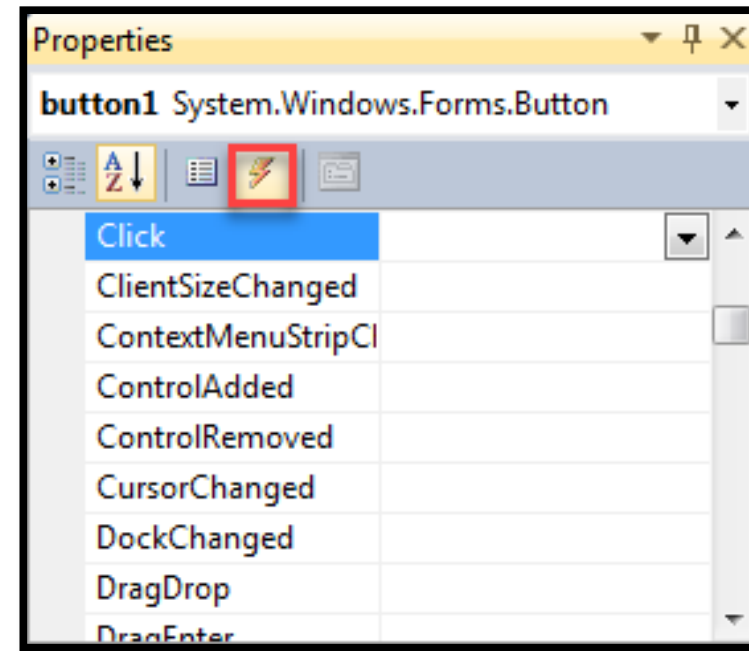
## Properties Windows



- Properties Windows คือหน้าต่างสำหรับจัดการ properties และ events ของ Control



properties



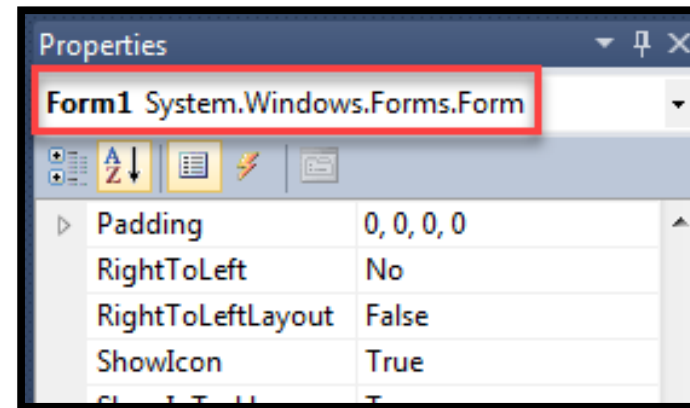
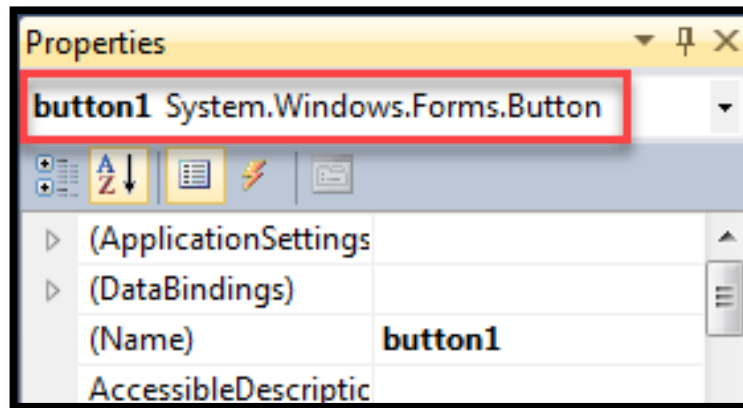
events



# Structure of Windows Form Application

## Properties Windows

- Properties Windows ของแต่ละ Control ก็จะมี properties และ events ที่ต่างกัน
- การดูว่า Properties ที่กำลังดูอยู่เป็นของ Control ไหน สามารถดูได้ที่



- ให้ทุกคน Focus ไปที่ button1

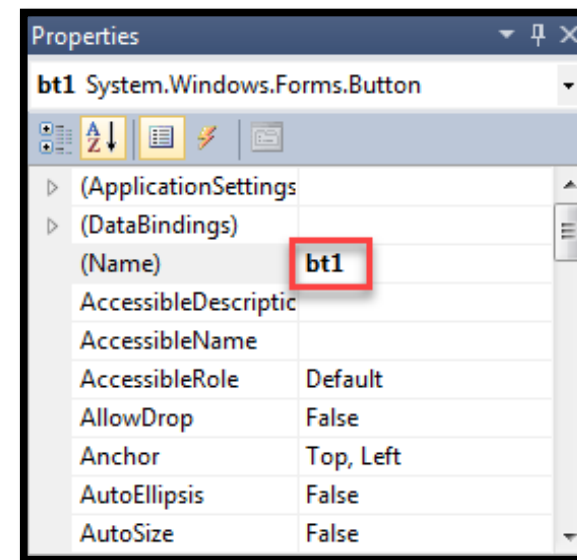
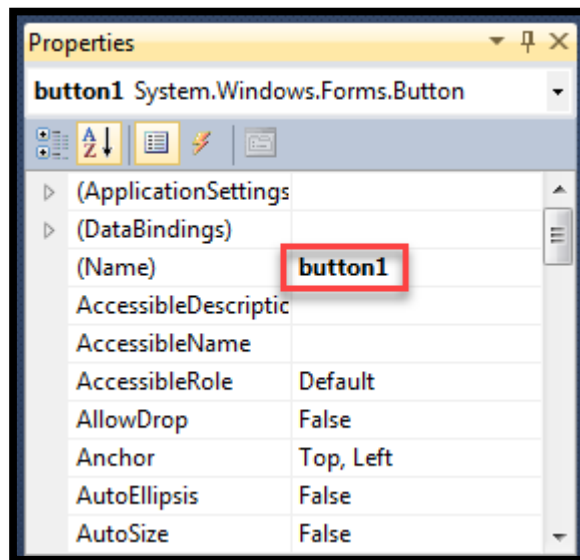




# Structure of Windows Form Application

## Properties Windows

- การเปลี่ยนชื่อ Control สามารถทำได้โดยแก้ไขที่ properties (Name)
- ให้ทุกคนเปลี่ยนชื่อจาก button1 ไปเป็น bt1



# Structure of Windows Form Application

## Properties Windows

- properties (Name) นี้เป็นการกำหนดชื่อของ Control ที่ใช้ในการเขียนโปรแกรม ด้วย ดังนั้นการเขียนโปรแกรมที่ใช้ Control จำนวนมาก ควรมีการกำหนดชื่อให้สามารถสื่อถึงบทบาทของ Control ได้





# Labels

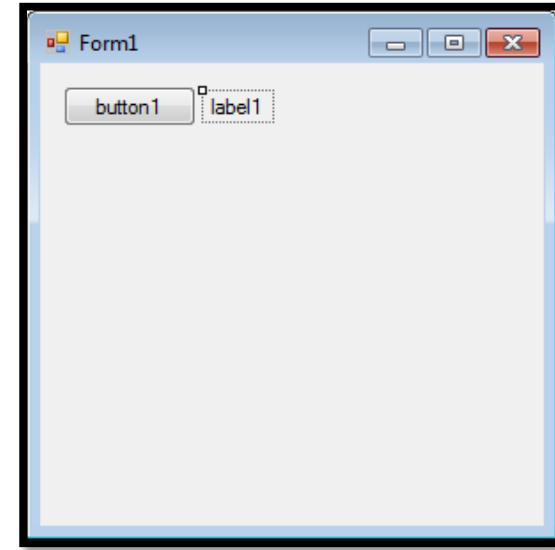
- Label เป็น control ที่เรียบง่ายที่สุด
- Label ใช้ในการแสดงข้อความบนหน้าจอ



# Labels

- ให้นักศึกษาลาก label จาก Toolbox มาใส่ใน Form1
- Properties ที่สำคัญของ label

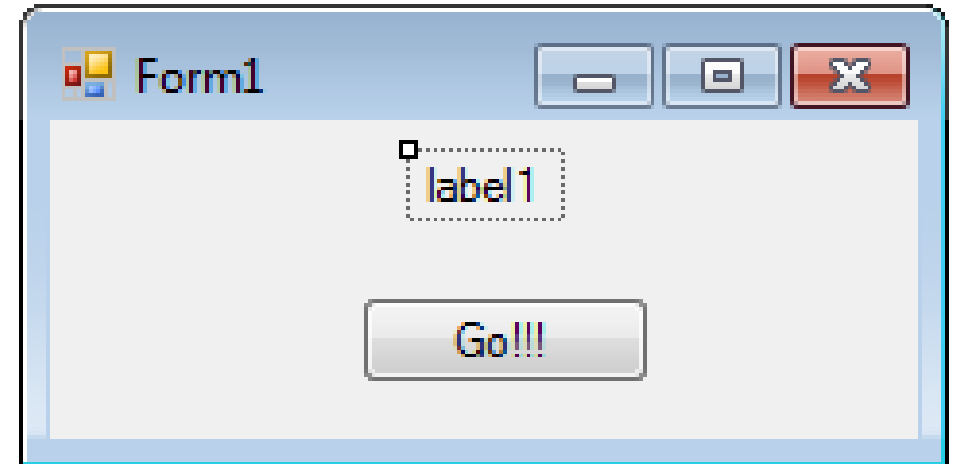
ชื่อ	การใช้งาน
(Name)	ชื่อของ Control
Location	กำหนดตำแหน่งของ label
Text	กำหนดข้อความของ label
Visible	กำหนดการมองเห็นของ label





# Labels

- ให้นักศึกษากำหนด Size ของ Form1 เป็น 240, 120
- กำหนด (Name) ของ label เป็น lb
- กำหนด Location ของ label เป็น 95, 10
- กำหนด Location ของ bt1 เป็น 80, 45
- กำหนด Text ของ bt1 เป็น Go!!!





# Button

- Button คือปุ่ม ซึ่งปุ่มนี้เน้นไปที่การคลิกเพื่อใช้งาน
- Button สามารถนำไปประยุกต์ใช้ได้หลายรูปแบบ



# Button

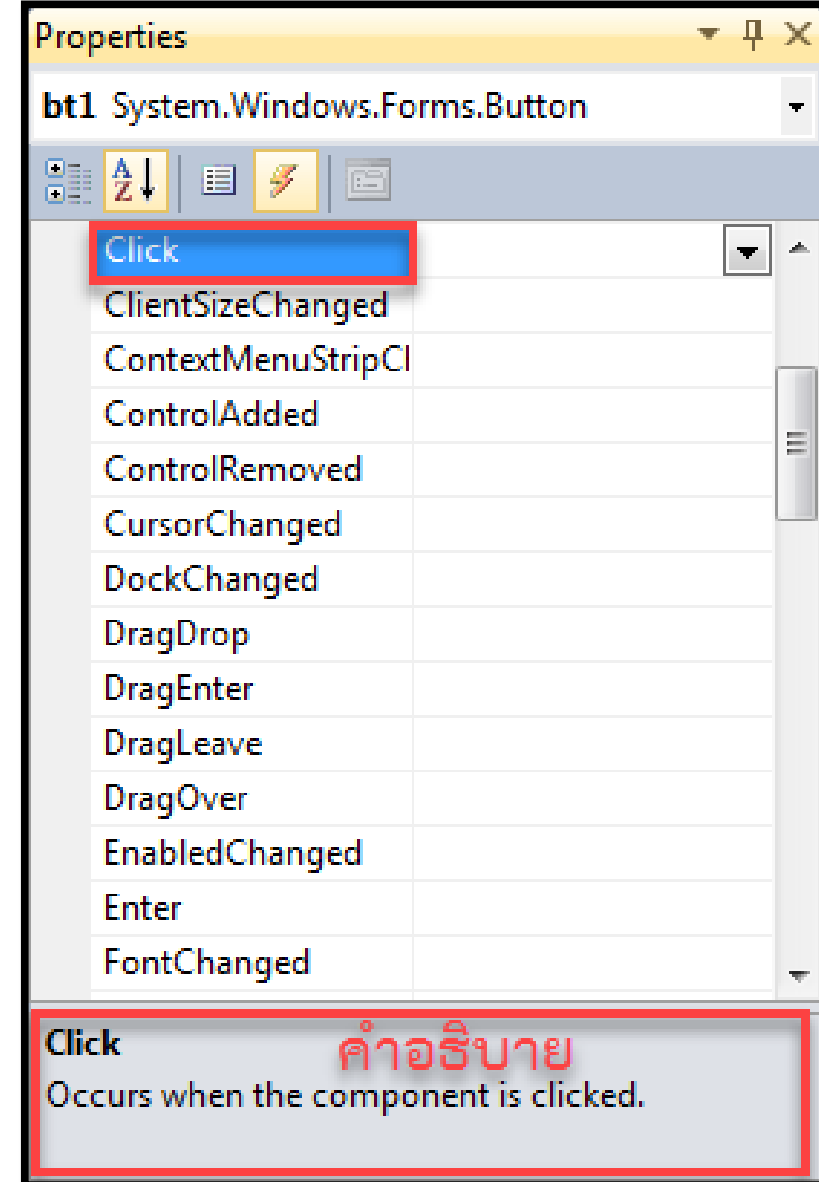


- Properties ที่สำคัญของ Button

ชื่อ	การใช้งาน
(Name)	ชื่อของ Control
Image	กำหนดรูปภาพใส่ไว้ใน Button
Location	กำหนดตำแหน่งของ Button
Text	กำหนดข้อความของ Button
Visible	กำหนดการมองเห็นของ Button

# Button

- Events ที่สำคัญของ Button คือ Click ซึ่งสามารถเรียกใช้ได้โดย ดับเบิลคลิกที่ Button ที่ต้องการ หรือไปที่ Properties Windows จากนั้นกดใช้ Events



Properties

bt1 System.Windows.Forms.Button

Click

ClientSizeChanged

ContextMenuStripCl

ControlAdded

ControlRemoved

CursorChanged

DockChanged

DragDrop

DragEnter

DragLeave

DragOver

EnabledChanged

Enter

FontChanged

**Click** คำอธิบาย  
Occurs when the component is clicked.







# Button

- ถ้าเราดับเบิ้ลคลิกที่ Button ไปแล้ว ที่ Form1.cs หรือ Control ที่บรรจุ Button อยู่ จะมีการสร้าง Method ชื่อ bt1\_Click เกิดขึ้น
- ภายใน Method bt1\_Click คือส่วนที่ให้เรา implement สิ่งที่จะให้เกิดเมื่อ Button ถูก Click

```
namespace chapter_5
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

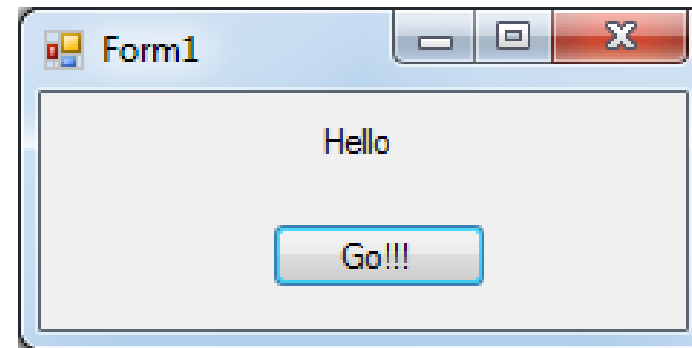
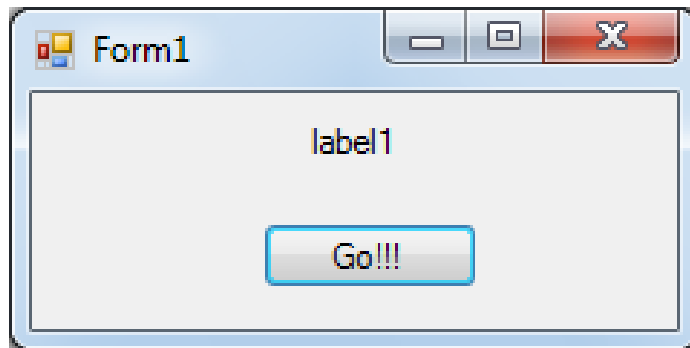
        private void bt1_Click(object sender, EventArgs e)
        {
        }
    }
}
```



# Button

- ยกตัวอย่างเช่น เมื่อ Click ที่ Bt1 แล้วให้ lb แสดงคำว่า Hello สามารถทำได้ดังนี้

```
private void bt1_Click(object sender, EventArgs e)
{
    lb.Text = "Hello";
}
```





# MessageBox

- MessageBox มันจะเป็นคำสั่งที่ใช้บ่อย เช่น แจ้งเตือน หรือแจ้งผลการทำงาน
- รูปแบบการใช้งาน MessageBox
  - `MessageBox.Show("Text", "Title Text", Type);`
- หรือใช้งานรูปแบบสั้นๆ ได้ดังนี้
  - `MessageBox.Show("Text")`



# MessageBox

- ตัวอย่างการใช้งาน MessageBox
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.AbortRetryIgnore);`
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.OK);`
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.OKCancel);`
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.RetryCancel);`
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.YesNo);`
  - `MessageBox.Show("Text", "Title", MessageBoxButtons.YesNoCancel);`
- ให้นักศึกษาลอง implement ดู



# MessageBox

- MessageBox ดังตัวอย่างนี้เป็นเพียงบางส่วนในการใช้งาน
- ในกรณีที่ MessageBox มี Button มากกว่า 1 ปุ่ม เราสามารถเขียนโปรแกรมเพื่อตรวจจับการกดแต่ละปุ่มได้ดังนี้

```
private void bt1_Click(object sender, EventArgs e)
{
    lb.Text = "Hello";
    if (MessageBox.Show("ยืนยันการออกจากระบบ", "แจ้งเตือนการออกจากระบบ", MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        MessageBox.Show("คุณเลือก Yes");
    }
    else
    {
        MessageBox.Show("คุณเลือก No");
    }
}
```

# TextBox



- คือกล่องข้อความที่มักใช้ในการรับข้อความจากผู้ใช้



# TextBox



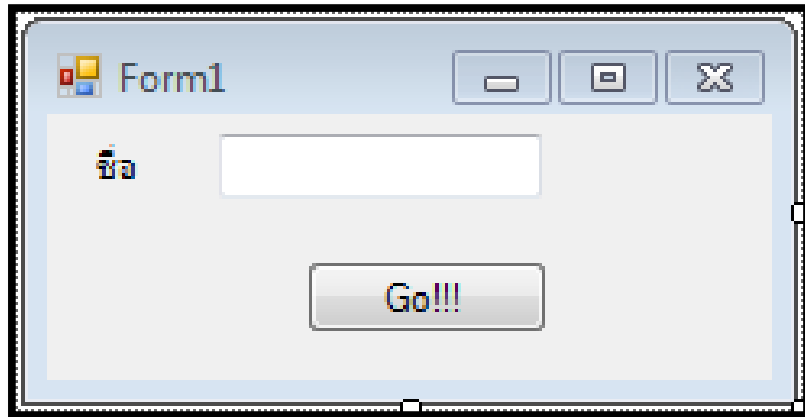
- Properties ที่สำคัญของ TextBox

ชื่อ	การใช้งาน
(Name)	ชื่อของ Control
Location	กำหนดตำแหน่งของ TextBox
Text	กำหนดข้อความของ TextBox
Visible	กำหนดการมองเห็นของ TextBox



# TextBox

- ตัวอย่างการใช้งานของ TextBox ได้ลองทำกันไปแล้ว ซึ่งมี Code ดังนี้



```
private void bt1_Click(object sender, EventArgs e)
{
    MessageBox.Show(textBox1.Text);
}
```



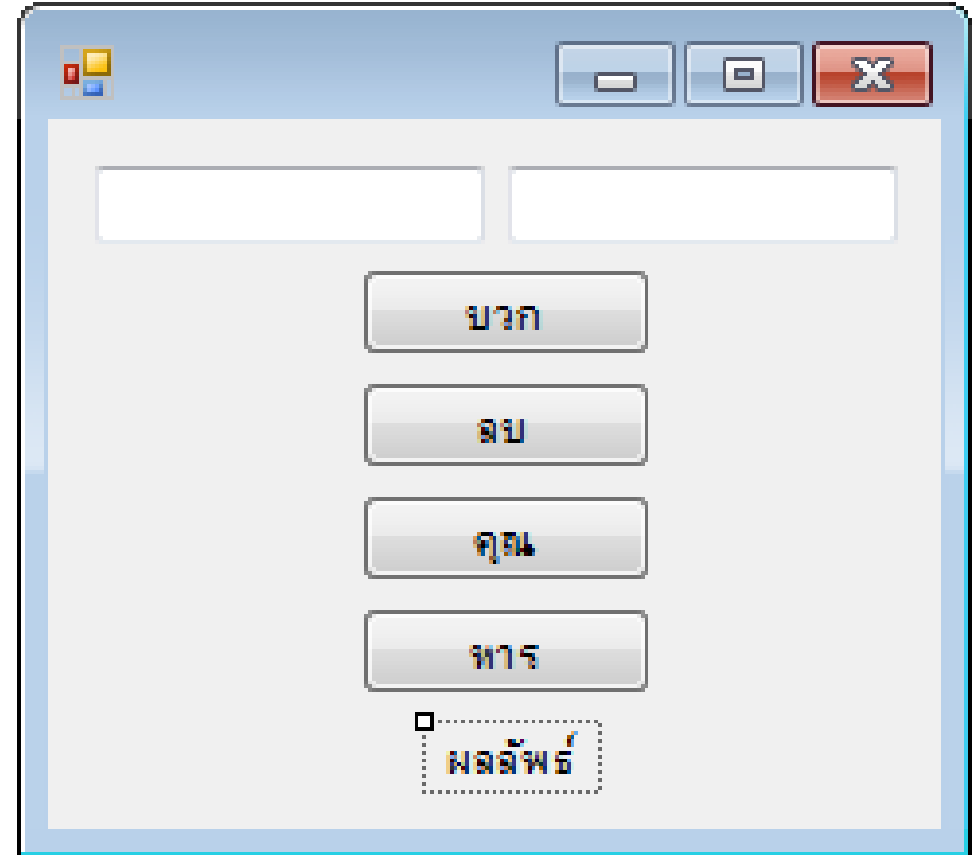
# Calculator



- ให้นักศึกษาออกแบบโปรแกรมโดยกำหนดให้มีลักษณะดังนี้
- Form1 กำหนด Size = 245, 220
- Textbox กำหนด (Name) = tb1
- tb1 กำหนด Location = 12, 12
- Textbox กำหนด (Name) = tb2
- tb2 กำหนด Location = 118, 12
- Button กำหนด (Name) = bt\_plus
- bt\_plus กำหนด Location = 80, 38
- bt\_plus กำหนด Text = บวก
- Button กำหนด (Name) = bt\_minus
- bt\_minus กำหนด Location = 80, 38
- bt\_minus กำหนด Text = ลบ
- Button กำหนด (Name) = bt\_multiply
- bt\_multiply กำหนด Location = 80, 96
- bt\_multiply กำหนด Text = คูณ
- Button กำหนด (Name) = bt\_divide
- bt\_divide กำหนด Location = 80, 125
- bt\_divide กำหนด Text = ทหาร

# Calculator

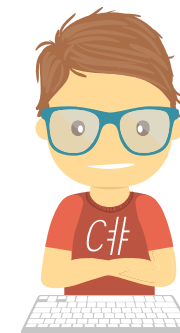
- สร้าง Label กำหนด (Name) = lb\_result
- lb\_result กำหนด Location = 99, 157
- lb\_result กำหนด Text = ผลลัพธ์



# Calculator



- เขียนโปรแกรมควบคุม Button ทั้งหมด เริ่มจาก bt\_plus โดยเมื่อ
- Click ที่ bt\_plus ให้นำค่าจาก tb1 และ tb2 มาบวกกันและแสดงผลที่ lb\_result
- Click ที่ bt\_minus ให้นำค่าจาก tb1 และ tb2 มาลบกันและแสดงผลที่ lb\_result
- Click ที่ bt\_multiply ให้นำค่าจาก tb1 และ tb2 มาบวกกันและแสดงผลที่ lb\_result
- Click ที่ bt\_divide ให้นำค่าจาก tb1 และ tb2 มาบวกกันและแสดงผลที่ lb\_result



สรุป