



ปฏิบัติการที่ 3 เรื่อง การเขียนโปรแกรมเชิงวัตถุด้วย Visual Studio VB.NET 2013

วิชา 322239 Database Application

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยขอนแก่น

จุดประสงค์ของบทเรียน

1. เพื่อสามารถเขียนโปรแกรมแกรมโดยใช้การเขียนโปรแกรมเชิงวัตถุ (OOP) โดยใช้ Visual Studio VB.NET ได้
2. เพื่อสามารถสร้าง Class และ Object โดยใช้ Visual Studio VB.NET ได้
3. เพื่อสามารถสร้างโปรแกรมคำนวณพื้นที่ เส้นรอบรูป และวาดรูปของรูปสี่เหลี่ยมจัตุรัส และวงกลม โดยใช้การเขียนโปรแกรมเชิงวัตถุ (OOP) ได้

การเขียนโปรแกรมเชิงวัตถุ

OOP หรือ Object Oriented Programming การเขียนโปรแกรมเชิงวัตถุ คือการเขียนโปรแกรมโดยการมองว่าส่วนประกอบของโปรแกรมเป็นเสมือนวัตถุชิ้นหนึ่งที่ประกอบไปด้วยคุณสมบัติ (Property) ซึ่งจะสามารถอธิบายได้ว่าวัตถุนี้คืออะไรและ วิธีการ (method) ซึ่งจะสามารถอธิบายพฤติกรรมของวัตถุนั้นว่าการเขียนโปรแกรมแบบ OOP เป็นการแบ่งซอฟต์แวร์หรือโปรแกรมออกเป็นส่วนๆ เรียกว่า Class โดยการนิยาม class และ object ก็เพื่อให้สามารถนำส่วนของซอฟต์แวร์หรือโปรแกรมส่วนนั้นกลับมาการทำงานของ class จะถูกกำหนดโดยส่วนอินเตอร์เฟสของ Method ส่วนการทำงานของส่วนที่เป็น Code จะไม่ถูกคำนึงถึงมากนักในการออกแบบภาษา OOP สนใจเฉพาะข้อมูลที่จะถูกประมวลผลมากกว่าฟังก์ชันที่ทำการประมวลข้อมูลนั้น ๆ

หลักการสำคัญของ OOP

- Class and Subclass
- Encapsulate
- Inheritance
- Polymorphism
- Abstract Data Type

Class

- คือกลุ่ม (category) ของ objects ที่มีคุณลักษณะและพฤติกรรมที่เหมือนกัน โดย class จะต้องประกอบไปด้วย data, behavior และ interface

- คือต้นแบบ (prototype) หรือพิมพ์เขียว ที่กำหนดตัวแปรและวิธีการ เพื่อนำไปใช้ได้ในทุก object

ของ class เช่น Class Square_class สี่เหลี่ยมจัตุรัสใน VB.NET

Objects

คือ สิ่งใดก็ตาม ซึ่งมีคุณลักษณะ (State) บ่งบอกถึงความเป็นตัวของมันเองในขณะนั้น และสามารถแสดงพฤติกรรม (Behavior) ของตัวเองออกมาได้ เช่น รถยนต์สีน้ำเงิน : มีความหมายคือ วัตถุประเภทรถยนต์ มีคุณลักษณะของสีเป็นสีน้ำเงิน และมีพฤติกรรมที่แสดงถึงการเคลื่อนที่ และหยุดได้หรือกล่าวได้ว่า object ก็คือข้อมูลของ class (เป็น entities ของ class) ซึ่งทุกๆ อย่างจะจัดเป็น objects โดยต้องประกอบไปด้วย

- ชื่อ (Identity)

- สถานะ (State) คุณลักษณะ หรือค่าของข้อมูล ซึ่งแทนด้วย Value

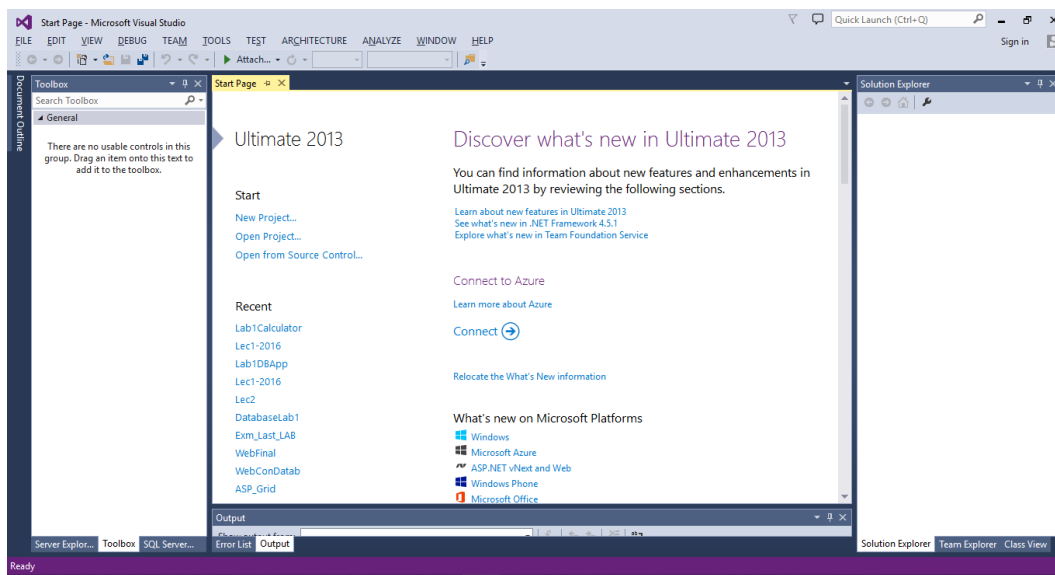
- พฤติกรรม (Behavior) ที่ระบุว่าสามารถทำอะไรได้บ้าง ซึ่งแทนด้วย Method

Method

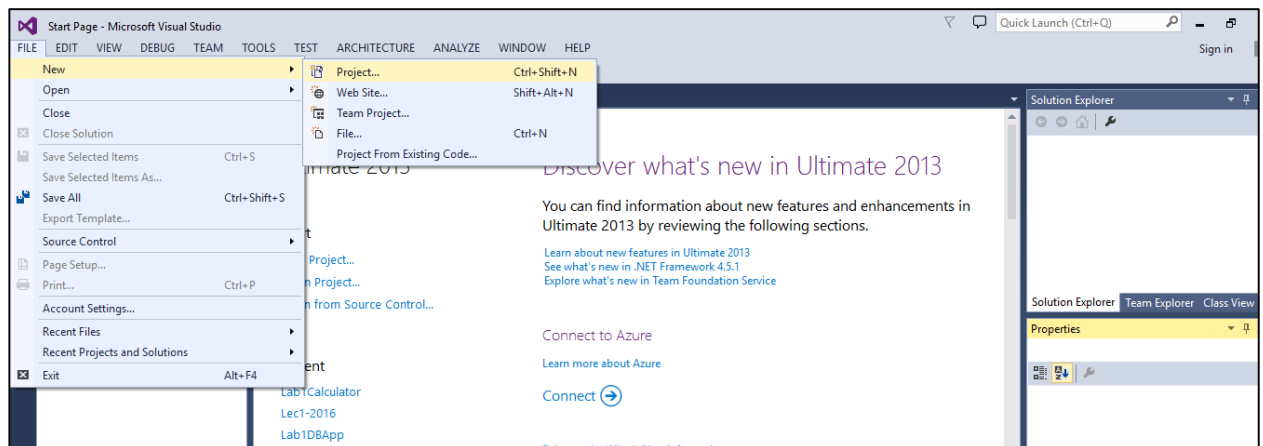
คือ function ที่บ่งบอกพฤติกรรมของ object ว่าทำอะไรได้บ้าง กำหนดไว้ใน class โดยต้องประกอบด้วยชื่อของ method เรียกว่า *Identifier* ตามด้วยเครื่องหมายวงเล็บ () โดยในวงเล็บอาจมี parameter list อยู่หรือไม่ก็ได้ เช่น Area(), Perimeter()

การสร้างโปรแกรมคำนวณพื้นที่ เส้นรอบรูป และวาดรูปของรูปสี่เหลี่ยมจัตุรัส

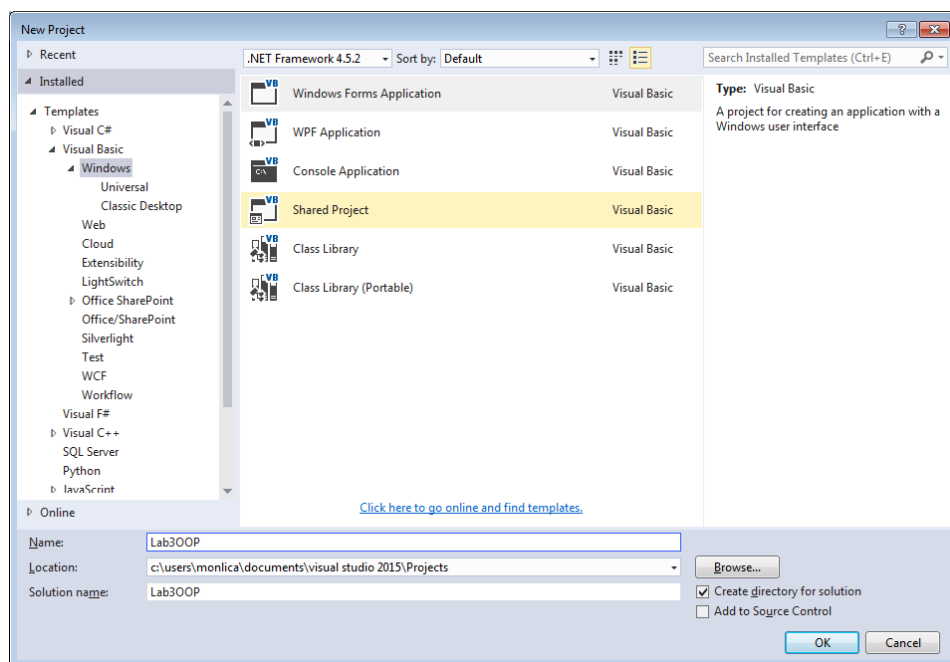
1. ให้นักศึกษาสร้าง Project ใหม่ขึ้น โดยเปิดโปรแกรม Visual Studio ขึ้นมาแล้ว ให้คลิกที่ New Project



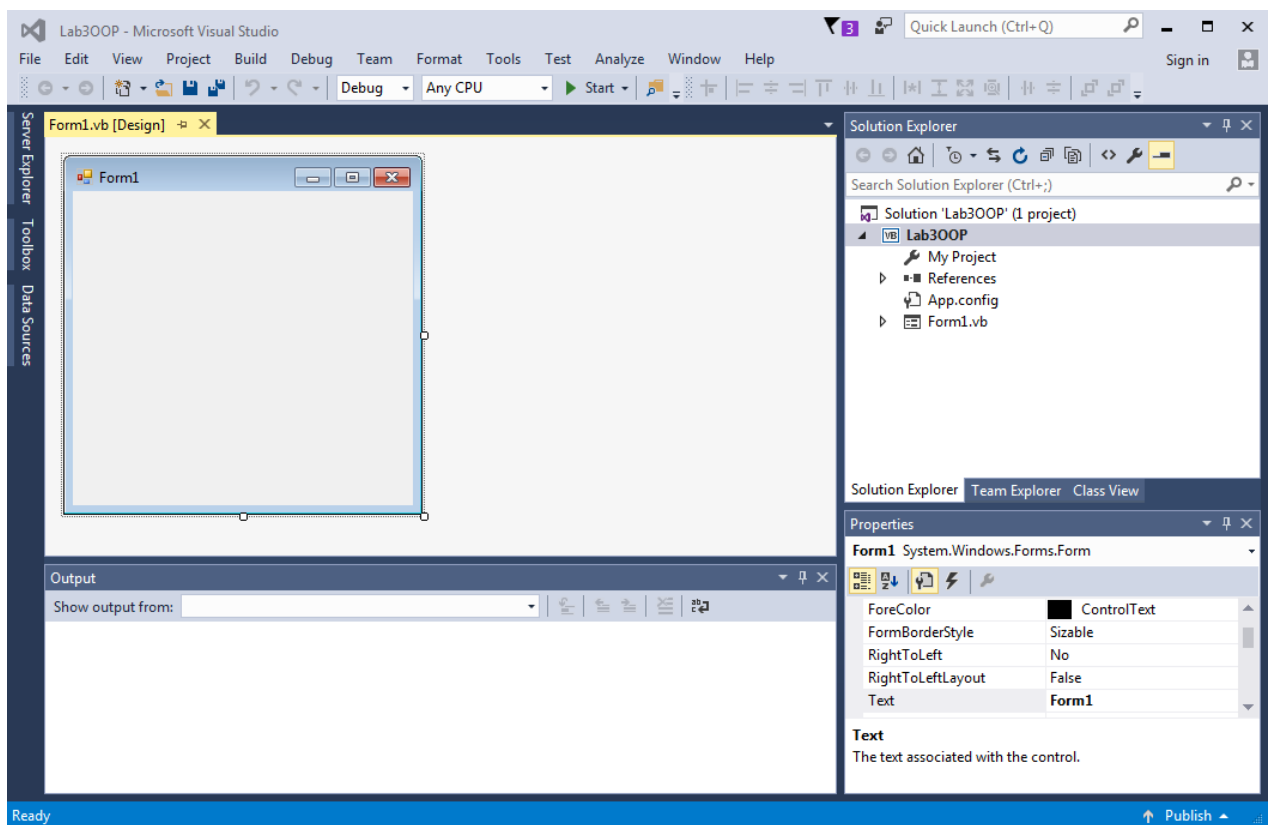
หรือสามารถสร้างโปรเจกใหม่ได้โดยไปที่เมนู File แล้วเลือก New จากนั้นคลิกที่ Project



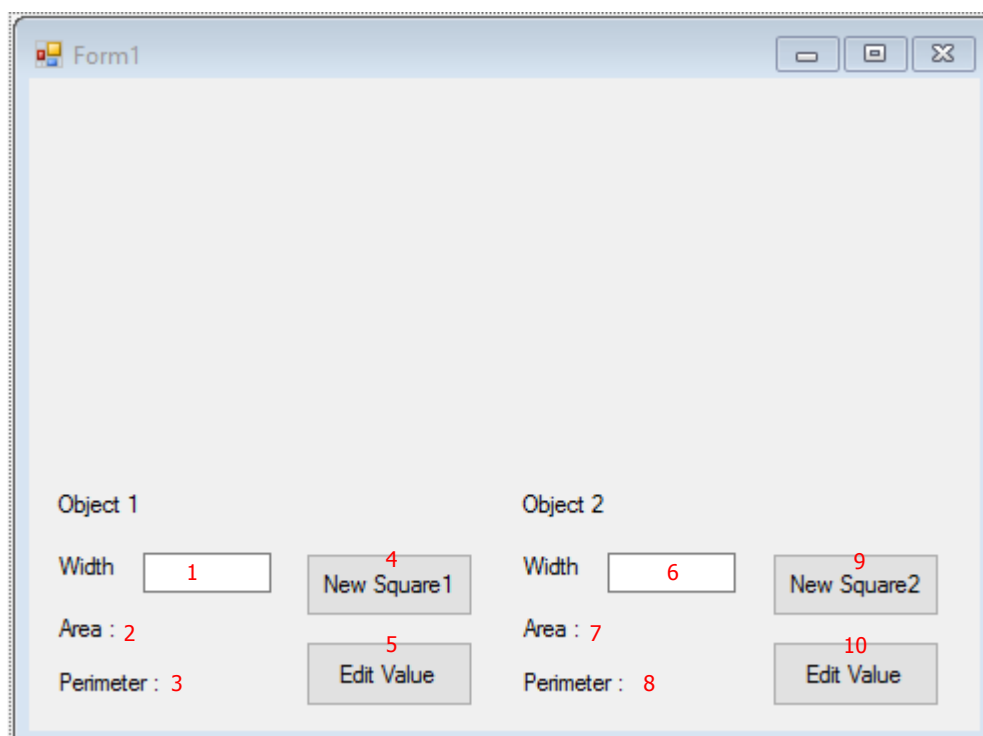
จากนั้นให้เลือก Windows Forms Application แล้วให้ตั้งชื่อเป็น Lab3OOP แล้วกดปุ่ม OK



จากนั้นจะปรากฏหน้าจอ ดังนี้



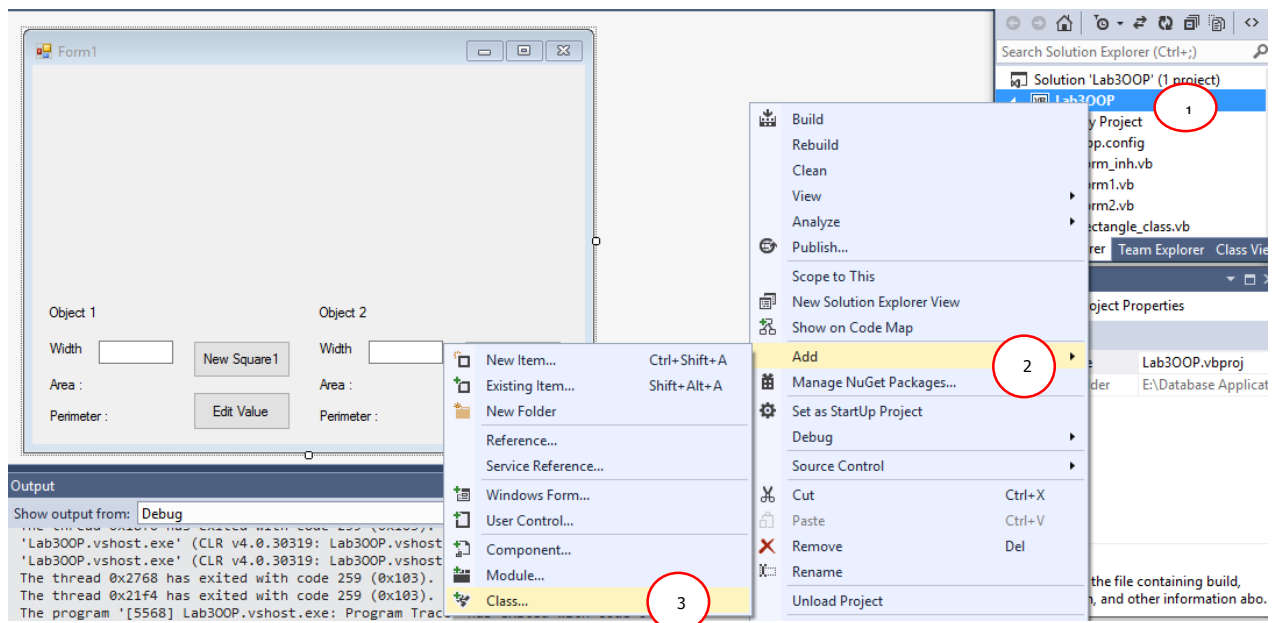
2. ให้นักศึกษาสร้างฟอร์มดังภาพด้านล่างนี้ และให้กำหนดค่าของ Control ตาราง



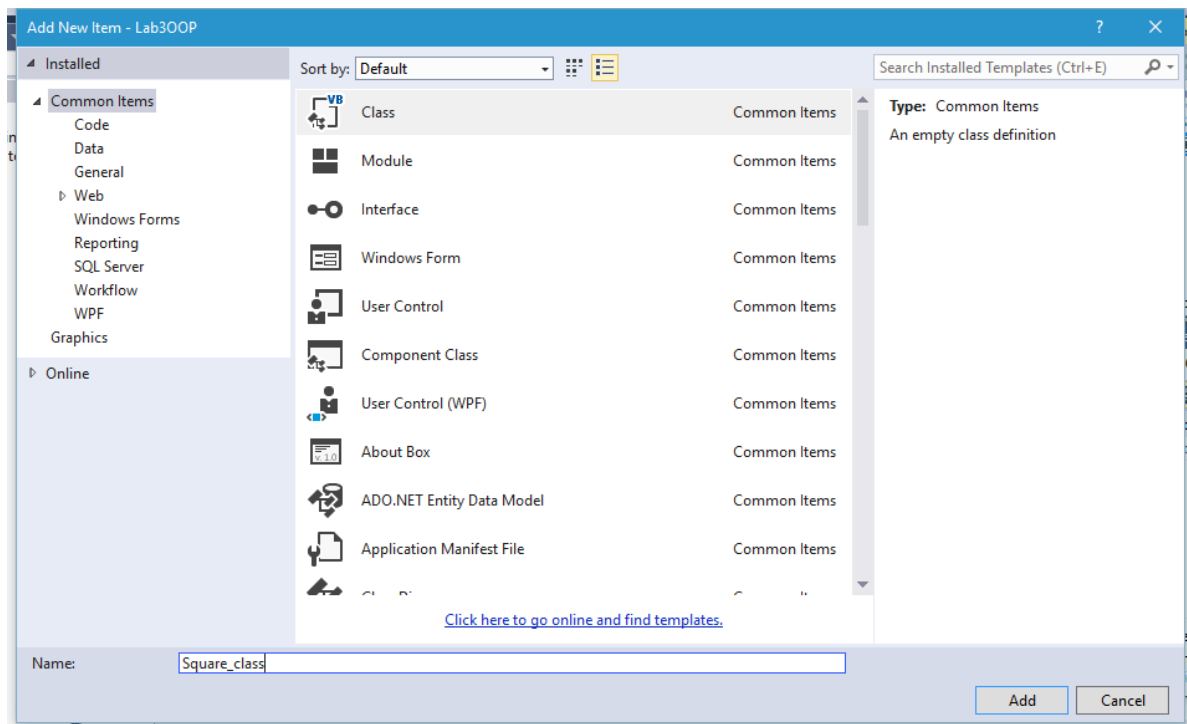
Object 1	Object 2
Width: <input type="text" value="1"/>	Width: <input type="text" value="6"/>
Area: 2	Area: 7
Perimeter: 3	Perimeter: 8
New Square1	New Square2
Edit Value	Edit Value

Toolbox	Property			
	Text	Design(Name)	Event	ทำงาน
1.Textbox		TxB_width1		
2.Label	Area :	Lb_square_area1		แสดงพื้นที่สี่เหลี่ยมจัตุรัส
3.Label	Perimeter :	Lb_square_perimeter1		แสดงเส้นรอบสี่เหลี่ยมจัตุรัส
4.Button	New Square1	btn_square1	click	คำนวณพื้นที่และเส้นรอบวงสี่เหลี่ยมจัตุรัส
5. Button	Edit Value	btn_edit1		
6.Textbox		TxB_width2		
7.Label	Area :	Lb_square_area2		แสดงพื้นที่สี่เหลี่ยมจัตุรัส
8.Label	Perimeter :	Lb_square_perimeter2		แสดงเส้นรอบสี่เหลี่ยมจัตุรัส
9.Button	New Square2	btn_square2	click	คำนวณพื้นที่และเส้นรอบวงสี่เหลี่ยมจัตุรัส
10. Button	Edit Value	btn_edit2		

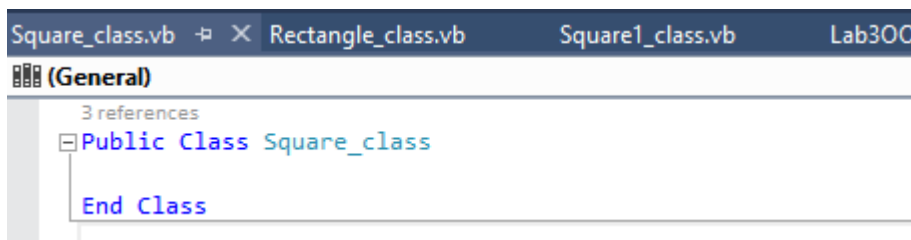
3. จากนั้นให้สร้าง Class โดยไปที่หน้าต่าง Solution Explorer คลิกขวาที่ Lab3OOP > Add > Class



4. ให้ใส่ชื่อ Class เป็น Square_class เลือก Add



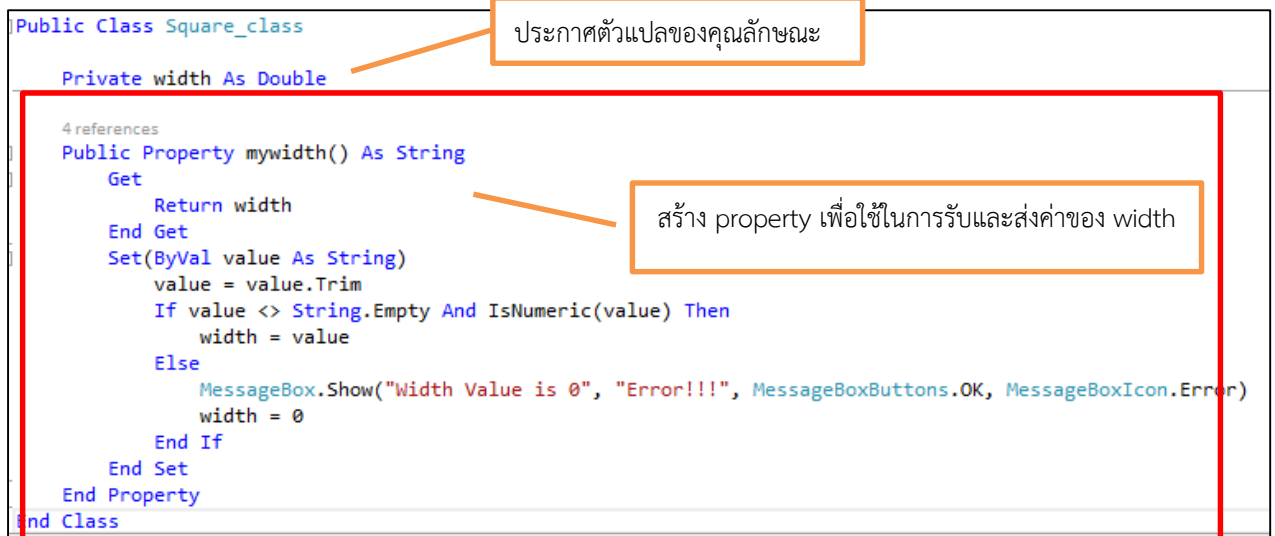
เมื่อคลิกปุ่ม Add แล้วจะปรากฏหน้าต่างดังภาพด้านล่าง เพื่อให้เพิ่มคำสั่งใน Class ของ รูปสี่เหลี่ยมที่ชื่อ Square_class



5. ให้นักศึกษาเพิ่มคำสั่งกำหนดคุณลักษณะในคลาส Square_class ที่สร้าง โดยกำหนดให้รับค่าผ่าน property ชื่อ mywidth ตามภาพด้านล่าง

Class: Square_class

คุณลักษณะ : ความกว้าง (width)



6. ให้นักศึกษาเพิ่ม Method ในคลาส Square_class ซึ่งมี 4 เมธอด ดังนี้

เมธอด :

1. Area เป็นฟังก์ชันรับค่าของ ความกว้าง(width) นำมาคำนวณหาค่าพื้นที่ของรูปสี่เหลี่ยมจัตุรัส
สูตรคำนวณ $Area = width * width$
2. Perimeter เป็นฟังก์ชัน รับค่าของ ความกว้าง(width) นำมาคำนวณหาค่าเส้นรอบรูปของรูปสี่เหลี่ยมจัตุรัส
สูตรคำนวณ $Perimeter = 4 * width$
3. Draw_Square เป็นโปรซีเยอร์สำหรับวาดภาพสี่เหลี่ยมจัตุรัส โดยใช้คำสั่ง DrawRectangle ของคลาส Graphics รูปแบบคำสั่ง ดังนี้
 $DrawRectangle(สีที่ใช้, จุดเริ่มต้น แกน x, จุดเริ่มต้น แกน y, ความกว้าง, ความยาว)$
4. Clear_Square เป็นโปรซีเยอร์สำหรับลบภาพสี่เหลี่ยมจัตุรัส โดยใช้คำสั่ง DrawRectangle ของคลาส Graphics เช่นกัน แต่จะเป็นการวาดสี่เหลี่ยมจัตุรัสทับ ด้วยสีของฟอร์มทำให้สี่เหลี่ยมจัตุรัสหายไป

โดยให้นักศึกษาคำสั่งของแต่ละ Method ดังนี้

<pre>Public Function Area(ByVal w As Double) As Double Dim area_Square As Double area_Square = w * w Return area_Square End Function</pre>	เมธอด Area : คำนวณหาพื้นที่ของสี่เหลี่ยมจัตุรัส
<pre>2 references Public Function Perimeter() As Double Dim perimeter_Square As Double perimeter_Square = (4 * width) Return perimeter_Square End Function</pre>	เมธอด Perimeter : คำนวณหาเส้นรอบวงของสี่เหลี่ยมจัตุรัส
<pre>2 references Public Sub Draw_square() Dim g_Draw As Graphics = Form1.CreateGraphics Dim Pen As Pen = New Pen(Color.Red, 3) g_Draw.DrawRectangle(Pen, 20, 20, Convert.ToInt32(mywidth), Convert.ToInt32(mywidth)) End Sub</pre>	เมธอด Draw_Square : วาดรูปสี่เหลี่ยมจัตุรัส
<pre>1 reference Public Sub Clear_square() Dim g_Clear As Graphics = Form1.CreateGraphics Dim Pen2 As Pen = New Pen(Color.FromArgb(Form1.BackColor.ToArgb), 3) g_Clear.DrawRectangle(Pen2, 20, 20, Convert.ToInt32(mywidth), Convert.ToInt32(mywidth)) End Sub</pre>	เมธอด Clear Square : ลบวาดรูปสี่เหลี่ยมจัตุรัส

7. เมื่อสร้าง Class เสร็จให้ไปที่ Form1 แล้ว ให้ดับเบิลคลิกที่ปุ่ม Btn_Square1 แล้วเขียนคำสั่งตามด้านล่าง

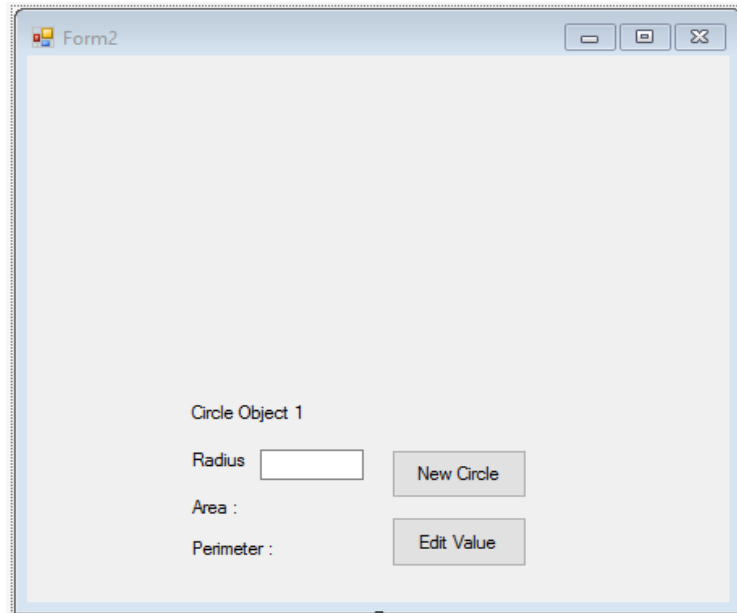
<pre>Dim MySquare1 As Square_class</pre>
<pre>Private Sub Btn_Square1_Click(sender As Object, e As EventArgs) Handles Btn_Square1.Click MySquare1 = New Square_class() MySquare1.mywidth = TxtB_width1.Text Lb_square_area.Text = "Area : " + CStr(MySquare1.Area(MySquare1.mywidth)) Lb_square_perimeter.Text = "Perimeter : " + CStr(MySquare1.Perimeter()) MySquare1.Draw_square() End Sub</pre>

8. หลังจากนั้น ให้ดับเบิลคลิกที่ปุ่ม Btn_edit1 แล้วเขียนคำสั่งตามด้านล่าง

<pre>Private Sub Btn_edit1_Click(sender As Object, e As EventArgs) Handles Btn_edit1.Click MySquare1.Clear_square() MySquare1.mywidth = TxtB_width1.Text Lb_square_area1.Text = "Area : " + CStr(MySquare1.Area(MySquare1.mywidth)) Lb_square_perimeter1.Text = "Perimeter : " + CStr(MySquare1.Perimeter()) MySquare1.Draw_square() End Sub</pre>

Assignment

1. จาก Form1 ให้นักศึกษาเขียนโปรแกรมสร้าง Object ที่ 2 ของ Class สีเหลี่ยมจัตุรัส ให้สามารถทำงานได้เหมือน Object ที่ 1 คือ หาพื้นที่ เส้นรอบรูป และวาดภาพของสีเหลี่ยมจัตุรัสได้ โดยเขียนโปรแกรมเชิงวัตถุให้สมบูรณ์
2. ให้สร้าง Form ใหม่ เพื่อสร้างหน้าจอดังนี้



และให้นักศึกษาสร้างคลาสวงกลม(Circle) ให้สามารถคำนวณหาพื้นที่ เส้นรอบวง และวาดรูปของวงกลมโดยเขียนโปรแกรมเชิงวัตถุ โดยมีสูตรคำนวณดังนี้

คุณลักษณะ : รัศมี (radius)

เมธอด : ** กำหนดค่า PI = 3.14

1. Area เป็นฟังก์ชัน รับค่าของ รัศมี (radius)นำมาคำนวณหาพื้นที่ของรูปวงกลม

สูตรคำนวณ $Area = PI * radius^2$

2. Perimeter เป็นฟังก์ชัน รับค่าของ รัศมี (radius)นำมาคำนวณหาเส้นรอบรูปของรูปวงกลม

สูตรคำนวณ $Perimeter = 2 * PI * radius$

3. Draw_Circle เป็นฟังก์ชันวาดภาพสีเหลี่ยมจัตุรัส โดยใช้คำสั่ง DrawEllipse ของคลาส Graphics รูปแบบคำสั่ง ดังนี้

DrawEllipse (สีที่ใช้, จุดเริ่มต้น แกน x, จุดเริ่มต้น แกน y, ความกว้าง, ความยาว)

สำหรับวงกลม ค่าความกว้างและความยาว จะมีขนาดเท่ากัน คือ รัศมี

4. Clear_Circle เป็นฟังก์ชันลบภาพสีเหลี่ยมจัตุรัส โดยใช้คำสั่ง DrawEllipse ของคลาส Graphics ซ้ำกัน แต่จะเป็นการวาดสีเหลี่ยมจัตุรัสทับด้วยสีของฟอร์มทำให้สีเหลี่ยมจัตุรัสหายไป

การส่ง Assignment

ให้นักศึกษา Zip ไฟล์โฟลเดอร์ที่เก็บไฟล์โปรแกรมไว้ โดยตั้งชื่อ LAB3_รหัสนักศึกษา.zip แล้วส่งเข้า LMS