

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

## แผนปฏิบัติการสอนสัปดาห์ที่ 2

วิชา 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม ระดับ: ปริญญาตรี

เรื่อง การใช้งานโปรแกรม และการเขียนโปรแกรมเบื้องต้น เวลา: บรรยาย 120 นาที

ปฏิบัติ 120 นาที

ก. วัตถุประสงค์การสอน	รายละเอียดตามที่จะพบไว้ใน
1. แนะนำโปรแกรม อุปกรณ์ และเครื่องมือต่าง ๆ	หน้า 18-33
2. เริ่มต้น และการบันทึก Project	
3. ตั้งค่า และใช้งานเมนูต่าง ๆ	
4. รู้จัก Components พื้นฐาน	
5. รู้จัก Object-Oriented Programming และ Object Pascal	

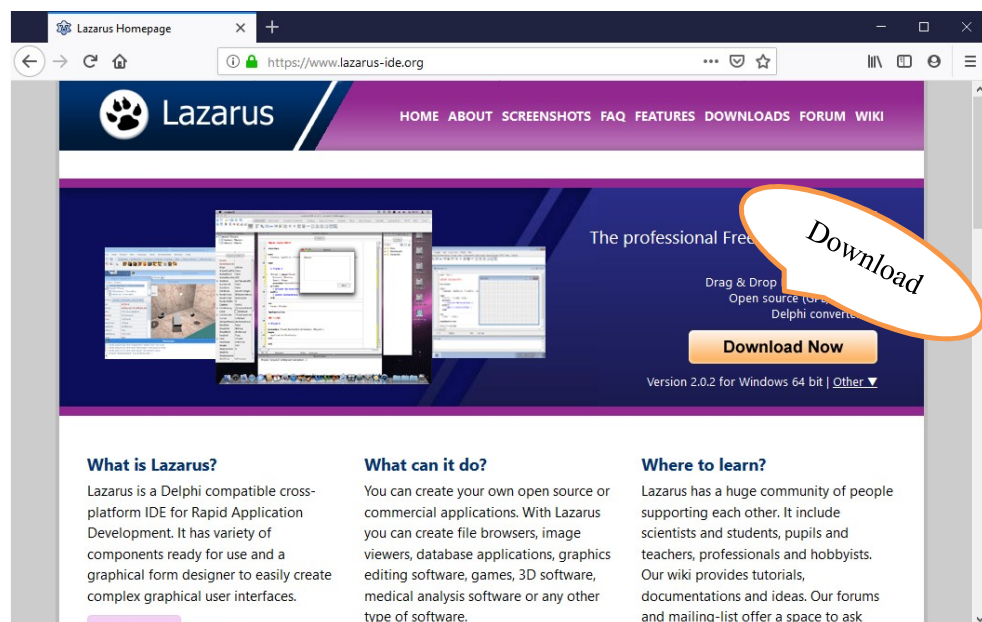
### ข. การจัดการเรียนการสอน

เวลา - นาที		0	60	120	180	240
วัตถุประสงค์		1, 2, 3, 4, 5				
การนำเข้าสู่บทเรียน						
ให้เนื้อหา						
สรุปเนื้อหา						
พัก						
ทำแบบฝึกหัด						
ให้เนื้อหา						
สรุปเนื้อหา						
ทดสอบและเก็บคะแนน						
ประเมินผล		พิจารณาจากผลการทดสอบในช่วงสุดท้าย				
วิธีการสอน:	บรรยาย					
	ถาม - ตอบ					
	ทำแบบฝึกหัด					
	บรรยาย และแสดงให้ดู					
	ทดสอบ					
สื่อการสอน:	คอมพิวเตอร์					
	Presentations					

## Content

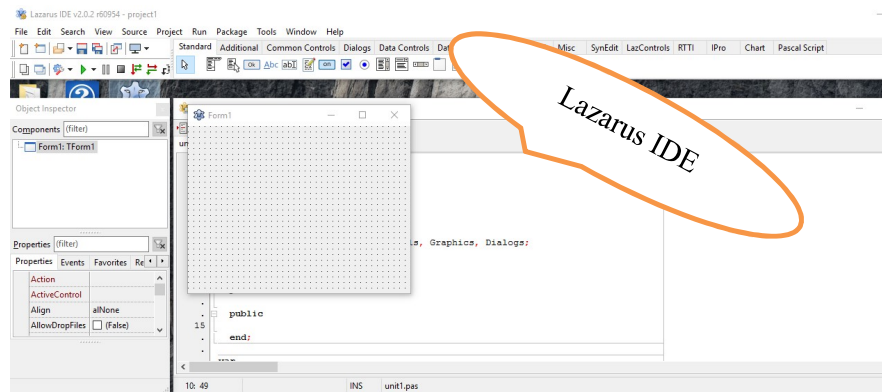
### 1. แนะนำโปรแกรม อุปกรณ์ และเครื่องมือต่างๆ

- ภาษาที่ใช้ในการเรียนวิชานี้คือ *Object Pascal*
- เครื่องมือพัฒนาโปรแกรมสามารถใช้ได้ 2 โปรแกรมคือ *Delphi* และ *Lazarus*
- ตัวอย่างทั้งหมดในเอกสารฉบับนี้สร้างขึ้นและทดสอบด้วยโปรแกรม *Delphi Community Edition* ซึ่งเป็น *Freeware* (เฉพาะ Version นี้เท่านั้น)
- สามารถนำ *Code* ไปใช้กับโปรแกรม *Lazarus* ได้ (*Freeware*)
- มีการใช้ *Component* เสริม นั้นหมายถึง *Code* บางส่วนเป็นคำสั่งเฉพาะไม่สามารถใช้งานทั่วไปได้ ต้องลง *Component* นั้นๆก่อนเสมอ หรือต้องทำการดัดแปลง *Code* เอง
- สำหรับโปรแกรม *Lazarus* สามารถ *Download* ได้ตามภาพด้านล่าง

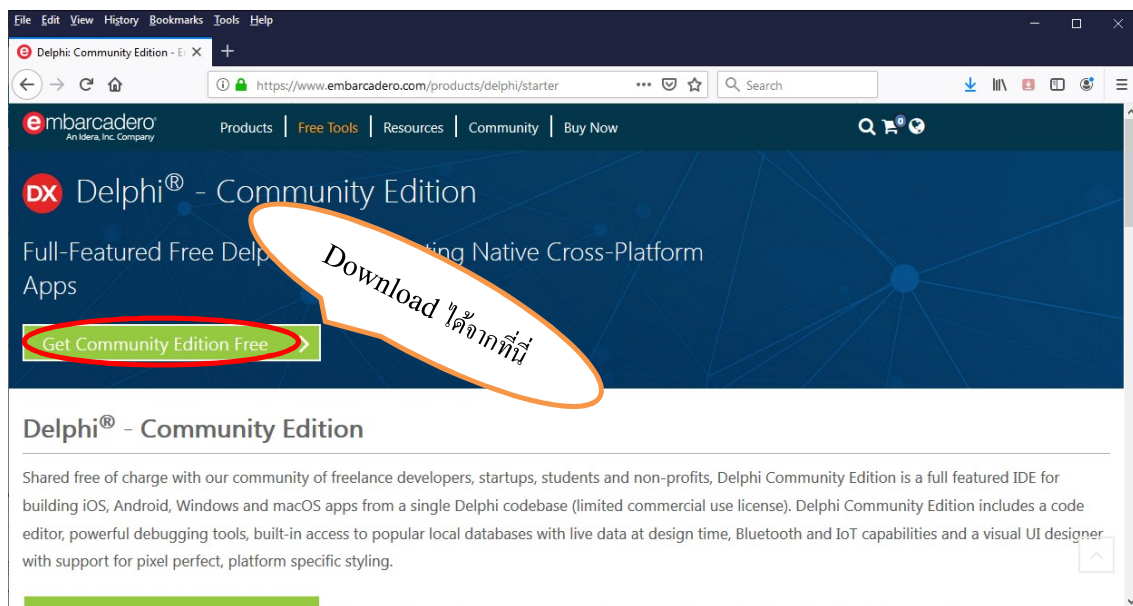


<https://www.lazarus-ide.org/>

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม



ให้ Download และ Install Delphi Community Edition ตามลำดับขั้นตอนด้านล่าง



<https://www.embarcadero.com/products/delphi/starter>

- ก่อน Download จะต้องลงทะเบียนก่อน หลังจากนั้นโปรแกรมจะส่งรหัสให้ทาง E-mail ให้ใช้รหัสนั้น Download โปรแกรม และ Register

Download the FREE Delphi Community

Already have an Embarcadero account? [Log in here](#)

☐ Are you a current customer? ☐ Are you a student?


First Name  Last Name

Email

Password  Verify Password

Company  Phone

United States  Select a State

☐ I'm not a robot 

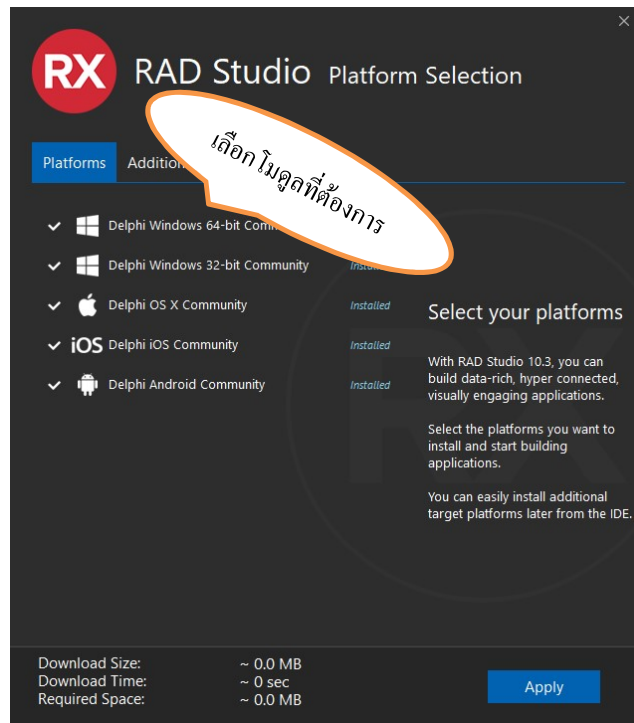
[Download Now](#)

กรอกรายละเอียด

<https://www.embarcadero.com/products/delphi/starter/free-download>

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

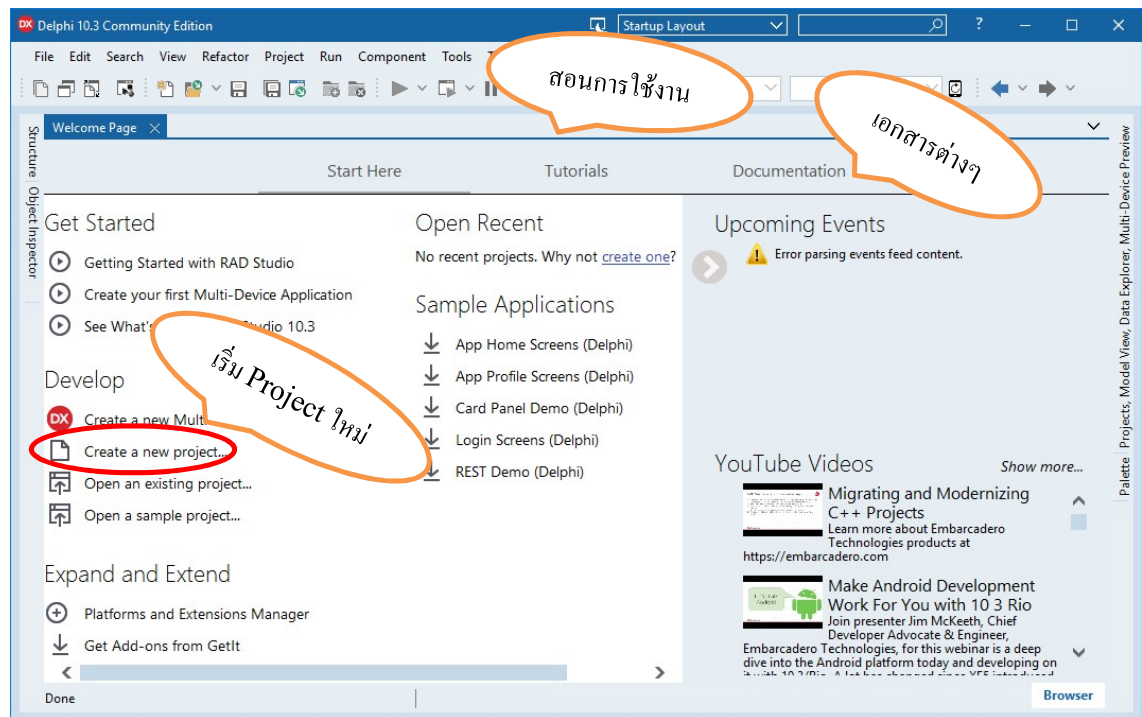
หลังจากสมัคร Account แล้วจะมี Email ส่งมาเกี่ยวกับข้อตกลง Serial Number และ Download URL ให้ทำการ Download Program มาติดตั้ง ในระหว่างติดตั้งจะมีการถามถึงรหัส License ให้กรอกรหัสที่ได้มาจาก Email



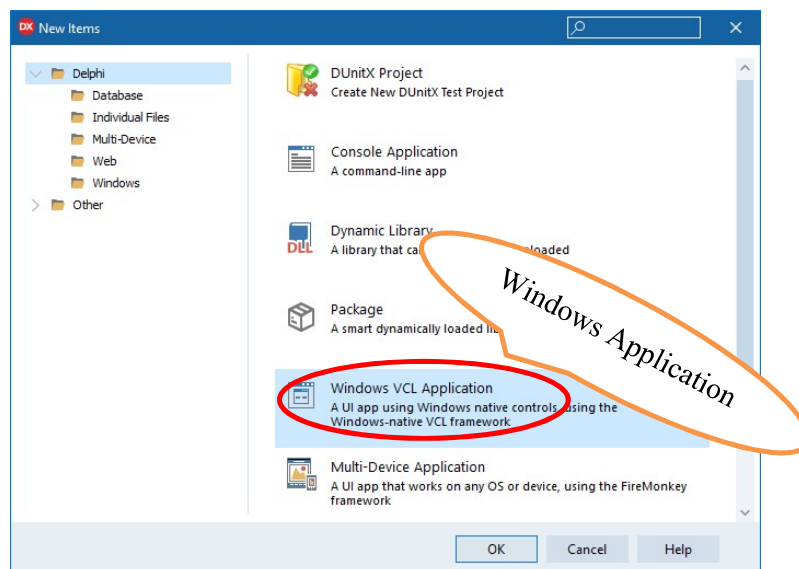
เนื้อหาในหลักสูตรครอบคลุมถึงการเขียนโปรแกรม VCL ที่ทำงานบนระบบ Windows และ Mobile Application ดังนั้นให้เลือกลง Program อย่างน้อย 2 Modules คือ Delphi Windows 32-bit Community และ Delphi Android Community

## 2. เริ่มต้น และการบันทึก Project

ในขั้นตอนแรกให้สร้าง Windows VCL Application ก่อน เพื่อสร้างโปรแกรมที่มีนามสกุล .exe ที่สามารถทำงานได้ในระบบปฏิบัติการ Windows ทั่วไปได้ มีขั้นตอนการสร้างดังนี้

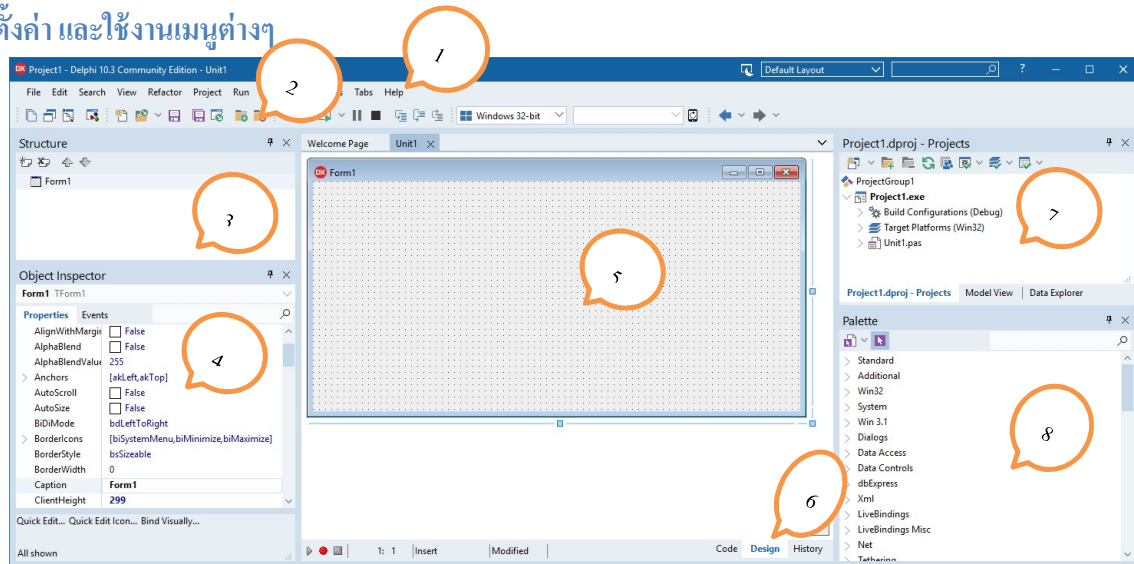


- กดที่ Create a new project



- เลือก Windows VCL Application

## 3. ตั้งค่า และใช้งานเมนูต่างๆ



เมื่อเลือกสร้าง Windows ACL Application แล้วโปรแกรมจะสร้าง Form, Code และ Environments ต่างๆให้ จากภาพด้านบนหน้าต่างโปรแกรมประกอบด้วย

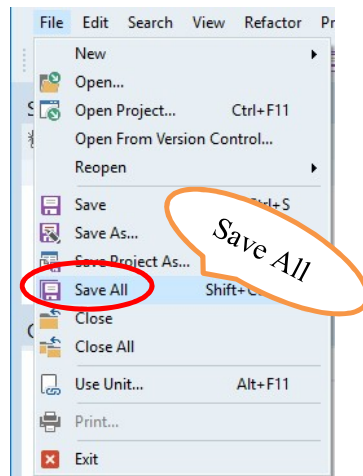
- (1) Main menu => เมนูใช้จัดการระบบต่างๆเช่นเปิด-บันทึกไฟล์, ตั้งค่าต่างๆ รวมถึงเรียกเครื่องมือที่เตรียมไว้
- (2) Tool bar => เป็นปุ่มกดคลิกเพื่อใช้งานคำสั่งต่างๆได้อย่างรวดเร็ว เช่น บันทึก Project, Run Program
- (3) Structure => ใช้แสดงว่าใน Program ประกอบไปด้วย Object อะไรบ้าง
- (4) Object Inspector => ใช้แสดง และตั้งค่า Properties ของ Object และกำหนด Event (ใน Event tab) ให้กับ Object นั้นๆ
- (5) Form designer => พื้นที่ออกแบบหน้าต่าง Program หรือ Graphic user interface (GUI) ใช้วาง Object ลงไป
- (6) Code and Design selector => Tab สำหรับเลือกการแสดงผลในโหมด Form หรือ Code
- (7) Project Tree view => แสดง และปรับตั้งค่ารายละเอียดต่างๆของ Project Group (ใช้เก็บหลายๆ Project)
- (8) Component Palette => ที่เก็บ Components เมื่อนำมาวางบน form จะถูกสร้างใหม่เป็น Object

หลังจากสร้าง Windows VCL Application แล้วให้ทำตามลำดับขั้นตอนนี้

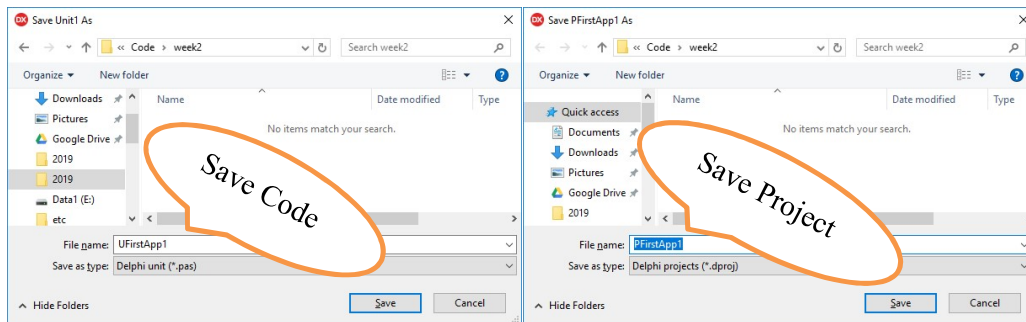
- 1 สร้าง folder สำหรับเก็บ Project
- 2 Files/Save All (Shift+Ctrl+S) => บันทึก Project ไปยัง Folder ที่สร้างไว้ โดยแยกบันทึก 2 ครั้งคือ Code (.pas) และ Project (.dproj)



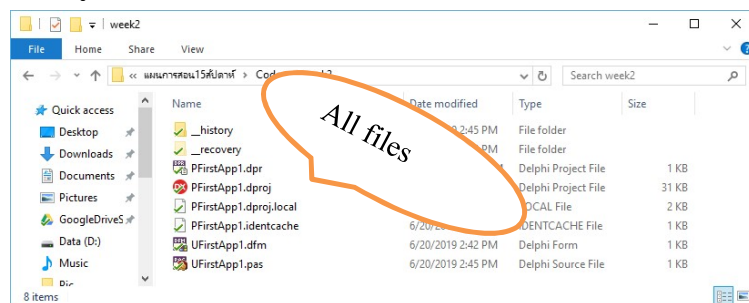
## 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม



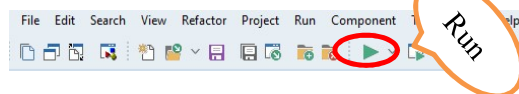
**Tip** การตั้งชื่อไฟล์ควรตั้งชื่อให้สอดคล้องกับหน้าที่ และควรมี Prefix U สำหรับ Unit และ P สำหรับ Project

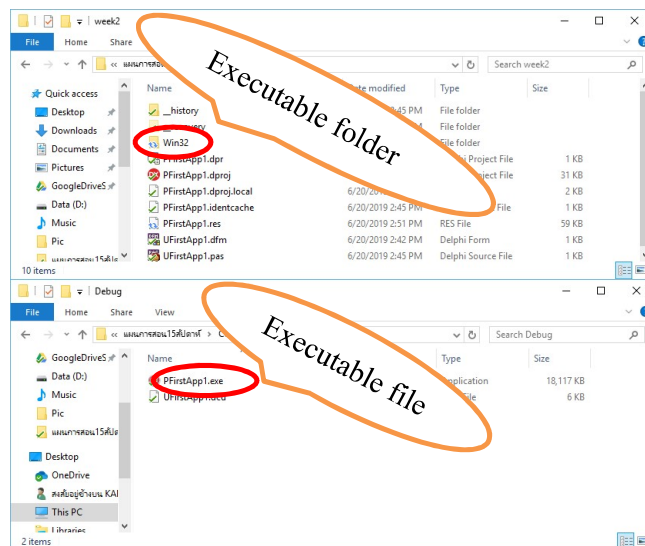


หลังจาก Save ทั้ง 2 ครั้งไปแล้ว Delphi จะสร้างไฟล์ที่จำเป็นต้องใช้เพิ่มเติมให้โดยอัตโนมัติ เช่น .dfm ใช้เก็บข้อมูลของ Form

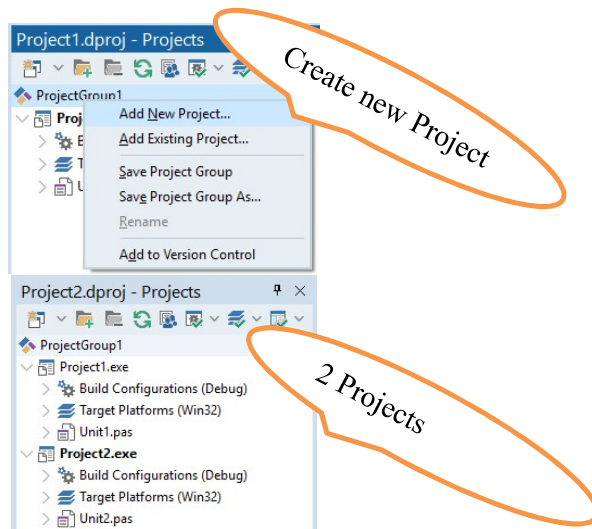


ให้ทดลองกดปุ่ม Run หรือกด F9 เพื่อเริ่มการทำงานของโปรแกรม Delphi จะสร้าง Folder Win32 ที่เก็บ Execute file .exe และ Delphi compiled unit .dcu ให้โดยอัตโนมัติ file.exe สามารถ copy ไปใช้งานบนระบบ Windows OS ได้เลยโดยตรง





การสร้าง Project โปรแกรมจะสร้าง ProjectGroup ให้โดยอัตโนมัติเราสามารถคลิกขวาที่ ProjectGroup1 เพื่อสร้างอีก Project ได้ นั่นหมายถึง 1 ProjectGroup สามารถมีได้หลาย Project ถ้าต้องการให้ Project ไหน Active ก็เพียงแค่ DbClick ที่ชื่อ Project.exe จากนั้นตัวอักษรจะเปลี่ยนไปเป็นตัวหนา นั่นหมายถึง Project นั้นกำลัง Active อยู่ ใน 1 ProjectGroup จะมีแค่ 1 Project ที่จะ Active ได้เท่านั้น



**Tip** Shortcuts สามารถดูได้จาก

[http://docwiki.embarcadero.com/RADStudio/Rio/en/Default\\_Keyboard\\_Shortcuts](http://docwiki.embarcadero.com/RADStudio/Rio/en/Default_Keyboard_Shortcuts)



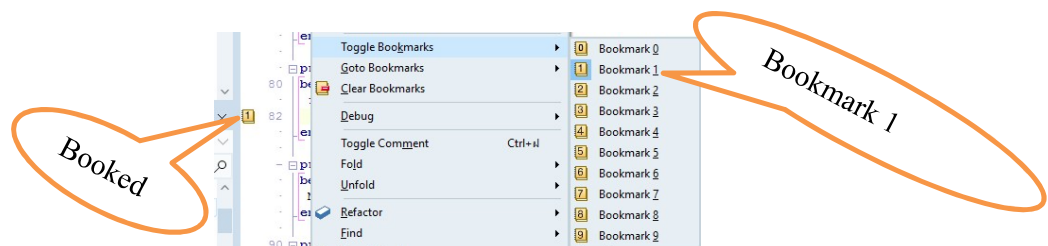
## ตัวอย่าง Shortcuts ที่ใช้บ่อย

Shortcut	Menu	Action
F1		Opens the help topic on a selected item.
Ctrl+S	File > Save	Saves only the module that is currently open in the IDE (and any associated header files).
Ctrl+Z	Edit > Undo Form Designer   right-click a component   Edit > Undo	Reverts the previous actions.
Ctrl+X	Edit > Cut Form Designer   right-click a component   Edit > Cut	Removes the current selection and stores it to the clipboard.
Ctrl+C	Edit > Copy Form Designer   right-click a component   Edit > Copy	Copies the current selection to the clipboard.
Ctrl+V	Edit > Paste Form Designer   right-click a component   Edit > Paste	Inserts the contents of the clipboard previously captured using Cut or Copy.
Ctrl+F / Ctrl+Q+F	Search > Find	Searches for the specified text in the active window.
Ctrl+R / Ctrl+H / Ctrl+Q+A	Search > Replace	Searches for the specified text and replaces it with other text.
F12	View > Toggle Form/Unit	Changes the view between the Form and the corresponding code unit.
F11	View > Object Inspector	Opens or focus on the Object Inspector.
F9	Run > Run	Compiles any changed source code and, if the compile is successful, executes your application, allowing you to use and test the application with the debugger built into the IDE.
F8	Run > Step Over	Tells the debugger to execute the next line of code. If the line contains a function, Step Over executes the function and then stops at the first line after the function.
F7	Run > Trace Into	Tells the debugger to execute the next line of code. If the line contains a function, Trace Into executes the function and then stops at the first line of code inside the function.
F4	Run > Run to Cursor	Executes the current program and stops at the cursor location.
Ctrl+Spacebar		Displays the Code Completion pop-up window or the Argument Value List.




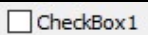
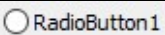
## 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

Shift+Ctrl+C		Invokes class completion for the class declaration in which the cursor is positioned.
Shift+Ctrl+I		Indents a selected block.
Ctrl+K+U / Shift+Ctrl+U / Shift+Tab		Outdents a selected line or block.
Ctrl+/ 		Adds or removes // to each line in the selected code block to comment the code.
Ctrl+K+1 (or 2-9)		Sets the bookmark.
Shift+Ctrl+1 (or 2-9)		Removes the bookmark.
Ctrl+0 (or 1-9) / Ctrl+Q+0 (or 1-9)		Goes to the corresponding bookmark.

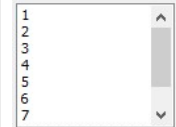
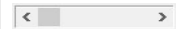

การสร้าง Bookmark (การบอกพิกัดบรรทัดใน Code เพื่อใช้ในการกระโดดไปตำแหน่งนั้น) ทำได้โดยคลิกขวาเลือก Toggle Bookmarks/Bookmark n (ctrl+shift+number[0-9]) หลังจากนั้นจะมีหมายเลข Bookmark ปรากฏด้านหน้า ถ้าต้องการไปยังตำแหน่ง Bookmark คลิกขวาเลือก Goto Bookmarks/Bookmark n (ctrl+shift+number[0-9])



#### 4. รู้จัก Components พื้นฐาน

Name	Image	Description	Properties	Events
TButton		ใช้เป็นปุ่มกด	Caption	OnClick
TLabel		แสดงผลตัวหนังสือ	Caption	OnClick
TEdit		รับค่าตัวหนังสือ	Text	OnChange
TMemo		รับค่าตัวหนังสือหลายบรรทัด	Lines	OnChange
TCheckBox		รับค่าการเช็ค	Checked	OnClick
TRadioButton		รับค่าการเช็ค(จัดกลุ่มได้)	Checked	OnClick

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

TListBox		แสดงรายการให้เลือก	Items	OnClick
TScrollBar		รับค่าตำแหน่งการเลื่อน	Position	OnChange
TPanel		เป็นที่วาง Components	Caption	OnClick

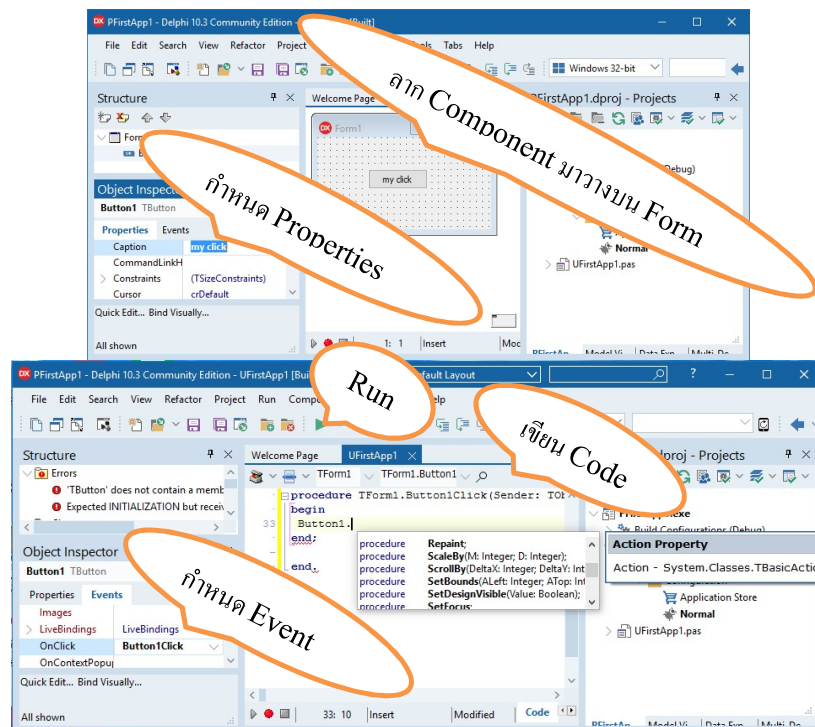
ตารางแสดงตัวอย่าง Component ที่ใช้บ่อย พร้อม Property and Event

เมื่อวาง Component ลงบน Form แล้ว Component จะถูกสร้างเป็น Object ที่มีทั้งคุณสมบัติเฉพาะ (Properties) และการตอบสนองต่อเหตุการณ์กระตุ้น (Event) เป็นของตัวเองอย่างอิสระ

## Tips

- ในบทนี้เป็นการศึกษา Component พื้นฐานเท่านั้นเพื่อให้นักศึกษาไม่เกิดความสับสน ถ้านักศึกษาต้องการศึกษาข้อมูลเพิ่มเติม สามารถหาข้อมูลได้จาก [http://docs.embarcadero.com/products/rad\\_studio/delphiAndcpp2009/HelpUpdate2/EN/html/delphivclwin32/contents.html](http://docs.embarcadero.com/products/rad_studio/delphiAndcpp2009/HelpUpdate2/EN/html/delphivclwin32/contents.html)
- ถ้าต้องการหาอะไรใน IDE เช่น Component ให้กด F6 แล้วพิมพ์ Keyword ที่ต้องการหา

## ขั้นตอนการเขียนโปรแกรม



## Tips

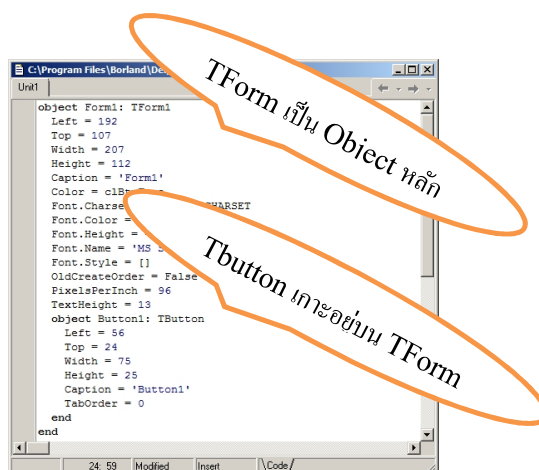
- // คือการทำ Comment บรรทัดเดียว
- { } คือการทำ Comment หลายบรรทัด
- (\* \*) คือการทำ Comment หลายบรรทัด
- ทุก Statement จะต้องปิดด้วย ;
- กลุ่มของคำสั่งต้องเริ่มต้นด้วย begin และปิดด้วย end;
- := คือการให้ค่าจากขวามาซ้าย เช่น a:=3;
- : คือการประกาศชนิดตัวแปรจากขวามาซ้าย เช่น a:Integer;

## 5. รู้จัก Object-Oriented Programming และ Object Pascal

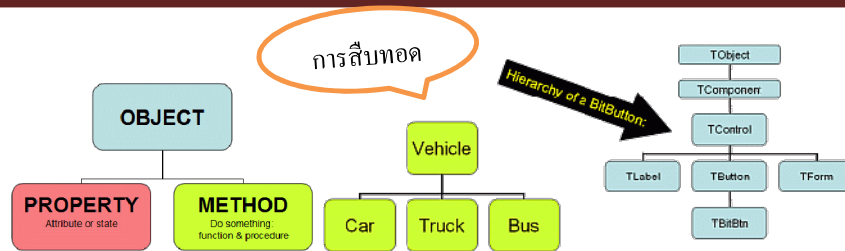
แนวคิดของ Object-Oriented Programming (OOP) นั้นคือการรวบรวมชุดคำสั่งที่ทำงานด้วยกันไว้เป็นก้อนเดียวกัน ประกอบด้วยข้อมูล (Properties) และชุดคำสั่งที่สามารถสั่งให้ทำงานได้ (Method) รวมไปถึงการตอบสนองต่อเหตุการณ์ต่างๆ (Event) ชุดคำสั่งนี้เมื่อนำไปใช้ในโปรแกรมจะเรียกว่า Object ใน 1 Program สามารถมีได้หลายๆ Objects ซึ่งแต่ละ Object ทำงานได้อย่างอิสระ ทุกๆ Object จะต้องเกาะอยู่บน Object หลักซึ่งในที่นี้คือ TForm

กลุ่มของชุดคำสั่งเหล่านั้นจะถูกเก็บรวบไว้ในโปรแกรม Delphi เรียกว่า Component เมื่อมีการนำมาวางลงบน Form จะมีการจัดสรรหน่วยความจำและทรัพยากรต่างๆให้ ณ ตอนนั้น Component จะถูกสร้างขึ้นใหม่กลายเป็น Object ซึ่งสามารถตั้งค่า Properties และเขียนโปรแกรมให้ตอบสนองต่อ Event ที่แตกต่างกันออกไปได้

อีกคุณสมบัติหนึ่งของ Object คือสามารถถ่ายทอดไปเป็นตัวใหม่ได้ เช่นเราสร้าง Vehicle Object ที่มีคุณสมบัติวิ่งได้บนถนนแล้ว เราก็สามารถเอา Vehicle มาปรับปรุงต่อยอดให้มีหลายล้อมีกะเบาะใช้ชนของได้กลายเป็น Truck Object ซึ่งเป็น Object ใหม่ที่สืบทอดมาจาก Vehicle Object นั้นเอง



# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม



<http://www.teachitza.com/delphi/oop.htm>

## ตัวอย่างการเขียน Object

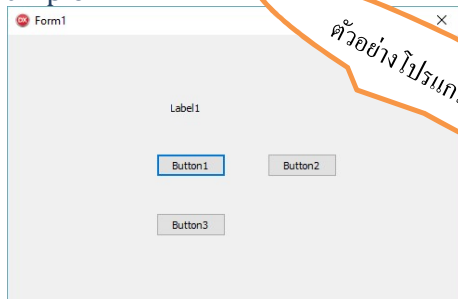
type

```
TClass_Name = class
  fFieldName1 : data_type;
  fFieldName2 : data_type;
  procedure Procedure_Name;
```

<http://www.teachitza.com/delphi/createclass.htm>

```
function Function_Name : data_type;
end;
var
  Object_Name : TClass_Nam
```

## Example



```
MAIN PROGRAM: unit UFirstApp1;
interface
uses
  Winapi.Windows, Winapi.Messages, System.SysUtils,
  System.Variants, System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, FireDAC.Stan.Intf,
  FireDAC.Stan.Option,
  FireDAC.Stan.Error, FireDAC.UI.Intf, FireDAC.Phys.Intf,
  FireDAC.Stan.Def,
  FireDAC.Stan.Pool, FireDAC.Stan.Async, FireDAC.Phys,
  FireDAC.Phys.MySQL,
```

```
FireDAC.Phys.MySQLDef, FireDAC.VCLUI.Wait,
Data.Win.ADODB, Data.DB,
FireDAC.Comp.Client, FireDAC.Stan.Param,
FireDAC.DatS, FireDAC.DApt.Intf,
FireDAC.DApt, Vcl.Grids, Vcl.DBGrids,
FireDAC.Comp.DataSet, Vcl.StdCtrls, UClass1;
```

```
class(TForm)
  Button1: TButton;
  Label1: TLabel;
  Button2: TButton;
  Button3: TButton;
  procedure Button1Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  procedure Button3Click(Sender: TObject);
private
  MyStudent:TStudent;
  { Private declarations }
public
  { Public declarations }
end;
var
  Form1: TForm1;
```

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

```
implementation
{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
    if MyStudent.setAge(15) then
        ShowMessage('SetAge OK');
    end;
    procedure TForm1.Button2Click(Sender: TObject);
    begin
        Label1.Caption:=IntToStr(MyStudent.getAge);
    end;
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    FreeAndNil(MyStudent);
    Application.Terminate;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
    MyStudent:=TStudent.Create;
end;
end.
```

คืนหน่วยความจำแล้ว  
ปิดโปรแกรม

สร้าง Object

```
unit UClass1;

interface

uses SysUtils;    //This is added to be able to use
IntToStr

type //Put structure of class in Interface section
    TStudent = class
        fName : String;
        fAge : Integer;
        function getAge: Integer;
        function setAge(Age:Integer):Boolean;
        constructor Create;
    end;
implementation
{ TStudent }
constructor TStudent.Create;
begin
```

```
inherited Create;
fName:="";
fAge:=0;
end;

function TStudent.getAge: Integer;
begin
    Result:=fAge;
end;

function TStudent.setAge(Age: Integer): Boolean;
begin
    if 0<Age then
        begin
            fAge:=Age;
            Result:=True;
        end
    else
        Result:=False;
    end;
end.
```

ทำการสร้างตามบรรทัด  
ก่อน

ใช้ Function ตรวจสอบ  
ความถูกต้องก่อน

เกิดตอนสร้าง Object

[Code\week2\ Ex\\_Class\\_1](#)

[Code\week2\ Ex\\_Class\\_2](#)

สร้าง Class TStudent2 ที่สืบทอดจาก Tstudent



```
unit UClass2;

interface

uses SysUtils, UClass1;    //This is added to be able to
use IntToStr

type    //Put structure of class in Interface section of
new unit
    TStudent2 = class(TStudent)

        fTall : Integer;
        function getTall: Integer;
        function setTall(Tall:Integer):Boolean;
        constructor Create;
    end;

implementation
```

```
{ TStudent2 }

constructor TStudent2.Create;
begin

end;

function TStudent2.getTall: Integer;
begin

end;

function TStudent2.setTall(Tall: Integer): Boolean;
begin

end;

end.
```

## Tip

- การเรียกใช้ Unit อื่นจะต้องประกาศชื่อ Unit ที่ถูกเรียกใช้ในส่วน uses ด้านบนเสียก่อน
- Double click ในช่องหลังชื่อ Event เป็นการ Assign Event ให้กับ Object นั้น โปรแกรมจะ Auto-gen Code ให้โดยอัตโนมัติ
- Object ทุกตัวควรตั้งชื่อใน Property Name ทันทีหลังจากวางลงบน Form และควรใช้ Prefix นำหน้าเสมอ
- การประกาศตัวแปร Procedure และ Function ใน Class ถ้าใส่ private ไว้ด้านบนหมายถึงเรียกใช้ได้เฉพาะใน Class ตัวเองเท่านั้น ถ้าใส่ public ไว้ด้านบนหมายถึงเรียกใช้ได้ทั้งใน Class ตัวเองและจากข้างนอก ถ้าไม่ใส่อะไรไว้ให้ถือว่าเป็น public
- ตัวแปรบางตัวใน Class ไม่ควรให้เข้าถึงได้โดยตรงจากภายนอกเพราะจะทำให้เกิดความผิดพลาด เช่น ค่าเปอร์เซ็นต์ไม่ควรเกิน 100 ดังนั้นจึงต้องให้ผู้ใช้เข้าถึงตัวแปรเหล่านั้นผ่านทาง Function ซึ่งทำหน้าที่รับค่ามารองความถูกต้องก่อนนำไปใช้แล้วคืนค่ากลับไปบอกว่าเปลี่ยนแปลงค่าสำเร็จไหม
- Type คือคำสั่งสร้างชนิดข้อมูลใหม่

## Exercise

- เขียนโปรแกรมที่มีปุ่มกด 2 ปุ่ม กดปุ่มแรกให้ Form เปลี่ยนเป็นสีแดง กดอีกปุ่มให้เปลี่ยนเป็นสีเขียว
- OOP หมายถึงอะไร
- แก้ Code สร้าง Class Tstudent2 ที่สืบทอดมาจาก Tstudent ให้มี Property fGender:String; และ Method SetGender กับ GetGender พร้อมเพิ่มปุ่ม 3 ปุ่มเพื่อ SetGender('Male'); GetGender('Female'); และ GetGender(); มาแสดงที่ Label
- สร้าง Class ชื่อ CarPark ที่ใช้เก็บข้อมูลรถในที่จอดรถโดยมีที่จอดทั้งหมด 100 คัน เมื่อรถเข้ามาจอดจะไปเรียกใช้ Function PushPark และลดจำนวนที่จอดคงเหลือไป 1 ที่ (ที่เหลือห้ามต่ำกว่า 0) เมื่อรถออกจะไปให้เรียกใช้ Function PullPark จำนวนที่จอดจะเพิ่มขึ้น 1 ที่ (ที่เหลือห้ามเกิน 100) สามารถเรียกดูที่ว่างได้จาก Function GetFreePark โดยเรียกใช้จากการกดปุ่ม และแสดงผลที่ Label บนโปรแกรมหลัก

## Assignment

- ทำรายงานเกี่ยวกับ Component ที่มีรายละเอียดของรากของ Class ว่าสืบทอดมาจากอะไร ใช้ทำอะไร มี Properties, Event and Method อะไร มาคนละ 5 ชนิดห้ามซ้ำกัน

## Answer Sheet

[Code\week2\Ans\\_Color\\_1](#)

[Code\week2\Ans\\_Class\\_1](#)