

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

## แผนปฏิบัติการสอนสัปดาห์ที่ 12

วิชา 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม ระดับ: ปริญญาตรี

เรื่อง การจัดการข้อมูล

เวลา: บรรยาย 120 นาที

ปฏิบัติ 120 นาที

ก. วัตถุประสงค์การสอน	รายละเอียดตามที่จะพบไว้ใน
1. Data Files	หน้า 166-181
2. Configurations File	
3. SQL Database และ Report	
4. ใช้งาน Free Web server, Dashboard และ Database	
5. IoT cloud และ Cloud MQTT	

### ข. การจัดการเรียนการสอน

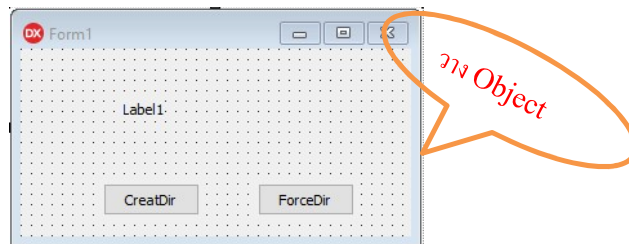
เวลา - นาที	0	60	120	180	240
วัตถุประสงค์	1, 2, 3, 4, 5				
การนำเข้าสู่บทเรียน					
ให้เนื้อหา					
สรุปเนื้อหา					
พัก					
ทำแบบฝึกหัด					
ให้เนื้อหา					
สรุปเนื้อหา					
ทดสอบและเก็บคะแนน					
ประเมินผล	พิจารณาจากผลการทดสอบในช่วงโมดสุดท้าย				
วิธีการสอน:	บรรยาย				
	ถาม - ตอบ				
	ทำแบบฝึกหัด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

## Content

### 1. Data Files

ระบบจัดเก็บข้อมูลของ Windows เป็นแบบแบ่งกลุ่มที่จัดเก็บข้อมูลเป็นกลุ่มย่อยเรียกว่า Folder ที่มี Drive เป็นส่วนใหญ่สุด ใน 1 Drive สามารถมีได้หลาย Folder ในแต่ละ Folder ก็ยังสามารถเก็บ Folder ย่อยๆได้อีกหลายอัน เส้นทางที่ใช้อ้างถึง Folder หรือFile ปลายทางนั้นเรียกว่า Path ตัวอย่างต่อไปนี้แสดงถึงวิธีการรับ Path ของ Program แล้วตรวจสอบว่ามี Folder xx อยู่ในนั้นไหม ถ้าไม่มีให้สร้างขึ้นใหม่

### Example



### Code

```
unit Unit1;

interface

uses
  Winapi.Windows, Winapi.Messages, System.SysUtils,
  System.Variants, System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Label1: TLabel;
  procedure FormCreate(Sender: TObject);
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
```

```
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
var
  P:String;
begin
  P:=ExtractFilePath(Application.ExeName)+'xx';
  if DirectoryExists(P) then
    ShowMessage(P + ' exists already')
  else
    begin
      if CreateDir(P) then
        ShowMessage('Directory created: ' + P)
      else
```

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

```
ShowMessage('Could not create ' + P);
end;
end;

procedure TForm1.Button2Click(Sender: TObject);
var
  P:String;
begin
  P:=ExtractFilePath(Application.ExeName)+'xx';
  if DirectoryExists(P) then
    ShowMessage(P + ' exists already')
  else
    begin
      ForceDirectories(P);
```

```
if DirectoryExists(P) then
  ShowMessage('Directory created: ' + P)
else
  ShowMessage('Could not create ' + P);
end;
end;

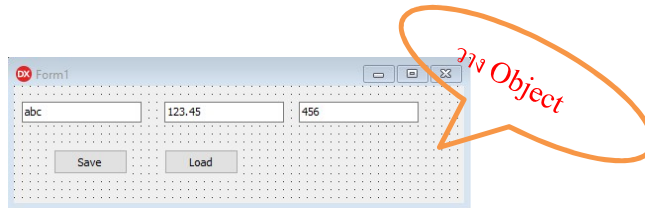
procedure TForm1.FormCreate(Sender: TObject);
begin
  Label1.Caption:=ExtractFilePath(Application.ExeName);
end;

end.
```

[Code\week12\ Ex\\_Dir\\_1](#)

การเขียนและอ่านไฟล์

## Example



## Code

```
unit Unit1;

interface

uses

  Winapi.Windows, Winapi.Messages, System.SysUtils,
  System.Variants, System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls;

Type
  TMyData = record
    MyName:String;
    MyFloat:Single;
    MyInt:Integer;
```

```
end;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
  procedure Button1Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  private
    MyData:TMyData;
    MyDataFile:TextFile;
```

```
MyDataPath:String;

procedure LoadMyData();
procedure DefaultMyData();
procedure SaveMyData();
{ Private declarations }
public
{ Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);
begin
begin
  MyData.MyName:=Edit1.Text;
  MyData.MyFloat:=StrToFloat(Edit2.Text);
  MyData.MyInt:=StrToInt(Edit3.Text);

  SaveMyData;
end;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  LoadMyData;
  Edit1.Text:=MyData.MyName;
  Edit2.Text:=FloatToStr(MyData.MyFloat);
  Edit3.Text:=IntToStr(MyData.MyInt);
```

```
end;

procedure TForm1.DefaultMyData;
begin
  MyData.MyName:='xxx';
  MyData.MyFloat:=1.23;
  MyData.MyInt:=12;

  SaveMyData;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin

  MyDataPath:=ExtractFilePath(Application.ExeName)+'My
  File.yyy';
end;

procedure TForm1.LoadMyData;
var
  Buff, Key, Data:String;
begin
  if FileExists(MyDataPath) then
    begin
      AssignFile(MyDataFile, MyDataPath); {Assigns the
      Filename}
      Reset(MyDataFile); {Opens the file for reading}
      try

        Key:='MyName = ';
        ReadLn(MyDataFile, Buff);
        Data:=Copy(Buff, Pos(Key, Buff)+Length(Key),
        Length(Buff) - Length(Key));
        MyData.MyName:=Trim(Data);

        Key:='MyFloat = ';
        ReadLn(MyDataFile, Buff);
```

```
Data:=Copy(Buff, Pos(Key, Buff)+Length(Key),
Length(Buff) - Length(Key));
MyData.MyFloat:=StrToFloat(Trim(Data));

Key:='MyInt = ';
Readln(MyDataFile, Buff);
Data:=Copy(Buff, Pos(Key, Buff)+Length(Key),
Length(Buff) - Length(Key));
MyData.MyInt:=StrToInt(Trim(Data));

except
  Closefile(MyDataFile); {Closes file}
  DefaultMyData;
  ShowMessage('Program Configuration Error Program
will Load Default!');
end;
Closefile(MyDataFile); {Closes file}
end
else
  DefaultMyData;
```

```
end;

procedure TForm1.SaveMyData;
var
  Buff:String;
begin
  AssignFile(MyDataFile, MyDataPath); {Assigns the
Filename}
  Rewrite(MyDataFile); {Create a new file }
  Buff:='MyName = '+MyData.MyName;
  Writeln(MyDataFile, Buff);
  Buff:='MyFloat = '+FloatToStr(MyData.MyFloat);
  Writeln(MyDataFile, Buff);
  Buff:='MyInt = '+IntToStr(MyData.MyInt);
  Writeln(MyDataFile, Buff);
  Closefile(MyDataFile); {Closes file }
end;

end.
```

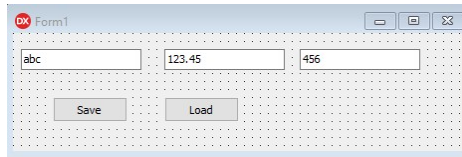
[Code\week12\ Ex\\_TextFile\\_1](#)

## 2. Configurations File

ใช้เก็บค่า Configuration ของโปรแกรมเพื่อใช้งานในครั้งถัดไปมีคำสั่งดังนี้

```
uses
  IniFiles
private
  IniFile : TIniFile;
- Load
  IniFile := TIniFile.Create(ChangeFileExt(Application.ExeName, '.ini'));
- Write
  IniFile.WriteString('SERVER', 'LAST', OpcServer.ServerName);
- Read
  ServerCombo.Text := IniFile.ReadString('SERVER', 'LAST', '');
- Destroy
  IniFile.Free;
```

## Example



วาง Object

## Code

```
unit Unit1;

interface

uses
  Winapi.Windows, Winapi.Messages, System.SysUtils,
  System.Variants, System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, IniFiles,
  Vcl.StdCtrls;

type
  TForm1 = class(TForm)
    Button2: TButton;
    Edit1: TEdit;
    ComboBox1: TComboBox;
    CheckBox1: TCheckBox;
    procedure FormCreate(Sender: TObject);
    procedure FormDestroy(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    iniFilePath:String;
    procedure ReadIniCfg();
    procedure WriteIniCfg();
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
```

```
implementation

{$R *.dfm}

procedure TForm1.Button2Click(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  iniFilePath:=ExtractFilePath(ParamStr(0)) + 'cfg.ini';
  if not FileExists(iniFilePath) then
    WriteIniCfg // Set Object Config as Default
  else
    ReadIniCfg;
end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  WriteIniCfg;
end;

procedure TForm1.ReadIniCfg;
var
  vIni:TIniFile;
begin
  vIni:=TIniFile.Create(iniFilePath);
  try
    Edit1.Text:=vIni.ReadString(Edit1.ClassName,
    'Edit1.Text', '');
  end;
```

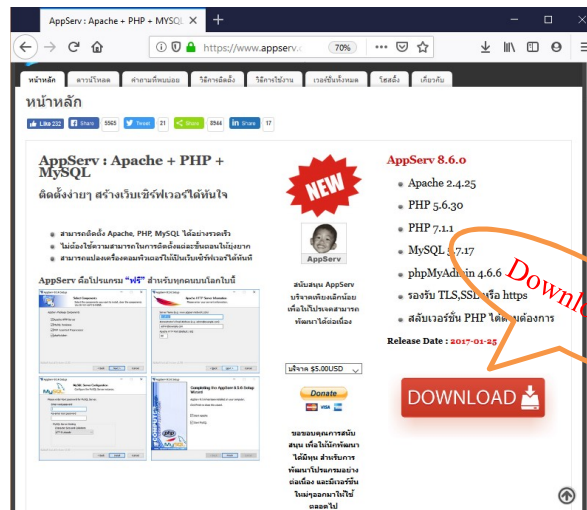
```
ComboBox1.ItemIndex:=vIni.ReadInteger(ComboBox1.Cl  
ssName, 'ComboBox1.ItemIndex', 0);  
CheckBox1.Checked:=vIni.ReadBool(CheckBox1.ClassNa  
me, 'CheckBox1.Checked', False);  
finally  
vIni.Free;  
end;  
end;  
  
procedure TForm1.WritelnCfg;  
var  
vIni:TIniFile;  
begin
```

```
vIni:=TIniFile.Create(iniFilePath);  
  
try  
vIni.WriteString(Edit1.ClassName, 'Edit1.Text',  
Edit1.Text);  
vIni.WriteInteger(ComboBox1.ClassName,  
'ComboBox1.ItemIndex', ComboBox1.ItemIndex);  
vIni.WriteBool(CheckBox1.ClassName,  
'CheckBox1.Checked', CheckBox1.Checked);  
finally  
vIni.Free;  
end;  
end;  
  
end.
```

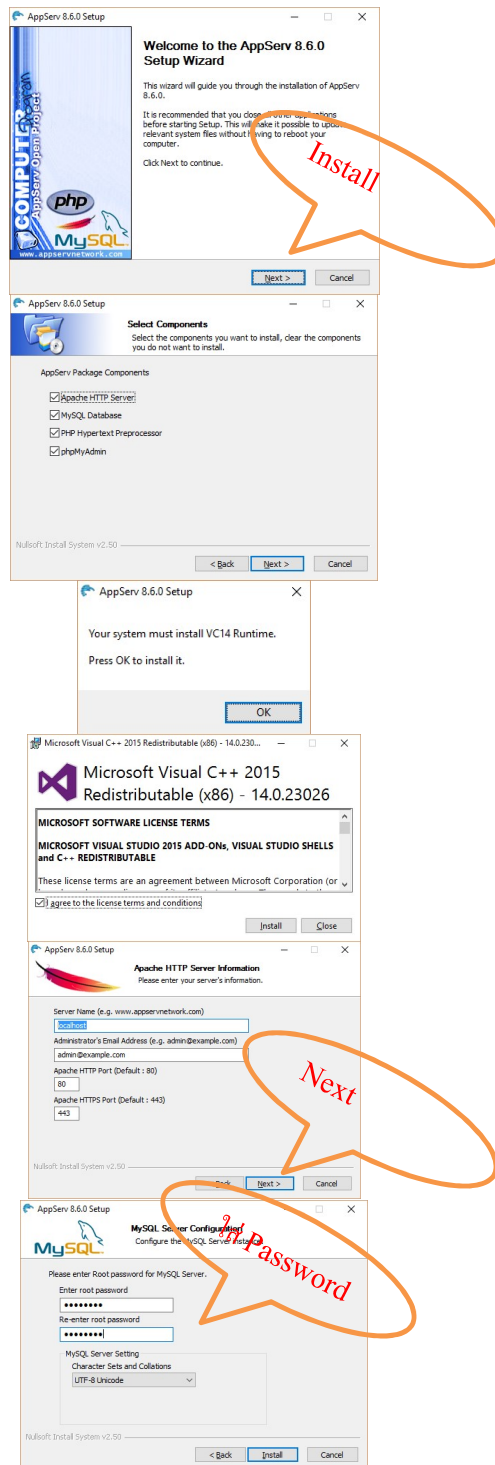
Code\week12\Ex\_ConfigFile\_1

### 3. SQL Database และ Report

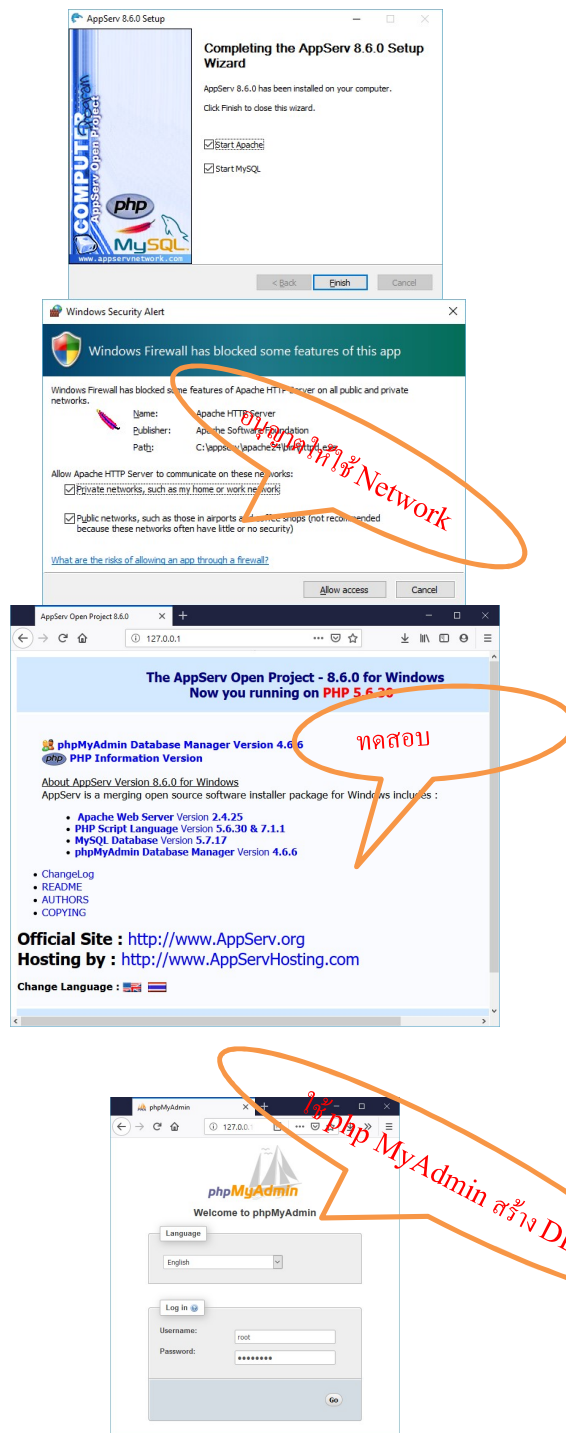
MySQL Database, Web Server และโปรแกรมช่วยต่างๆถูกรวมไว้ใน Package เดียวกันให้สามารถ Download ได้ฟรี เรียกว่า Appserv สามารถ Download ได้จาก URL ด้านล่าง (ใช้ได้เฉพาะ 32 บิต)



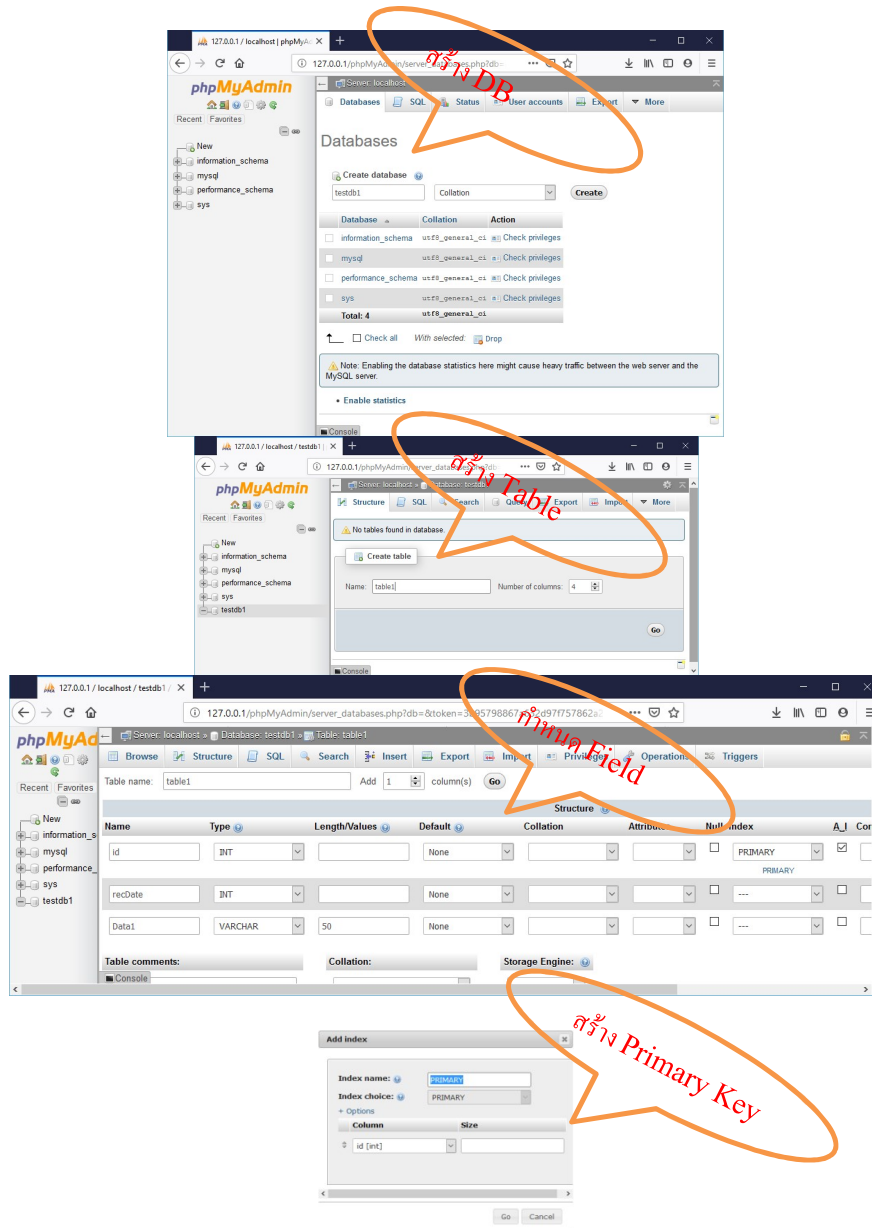
<https://www.appserv.org/th/>



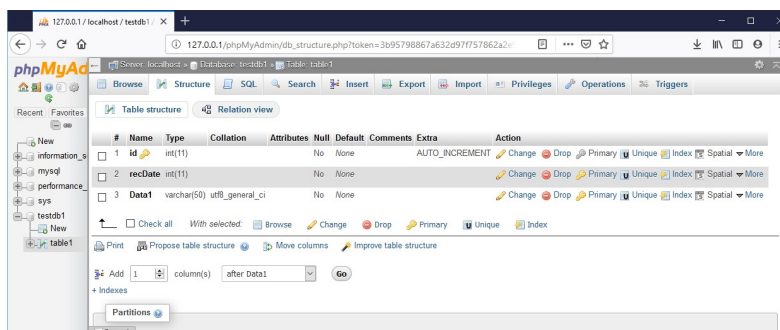




# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม



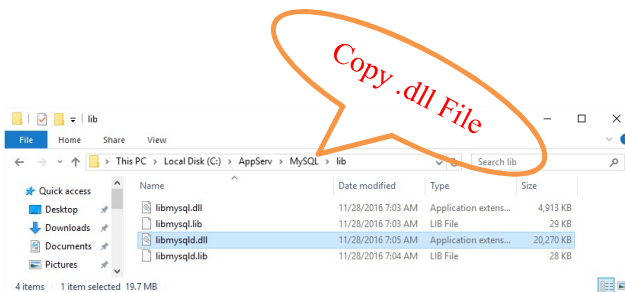
กดปุ่ม Save ด้านล่าง



[http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Connect\\_to\\_MySQL\\_Server\\_%28FireDAC%29](http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Connect_to_MySQL_Server_%28FireDAC%29)

# 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

## Example



[Copy](#)

C:\AppServ\MySQL\lib\libmysql.dll

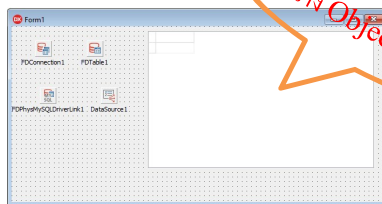
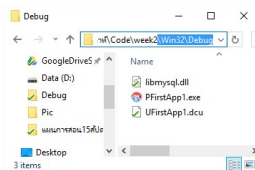
C:\AppServ\MySQL\lib\libmysql.dll

[Copy to](#)

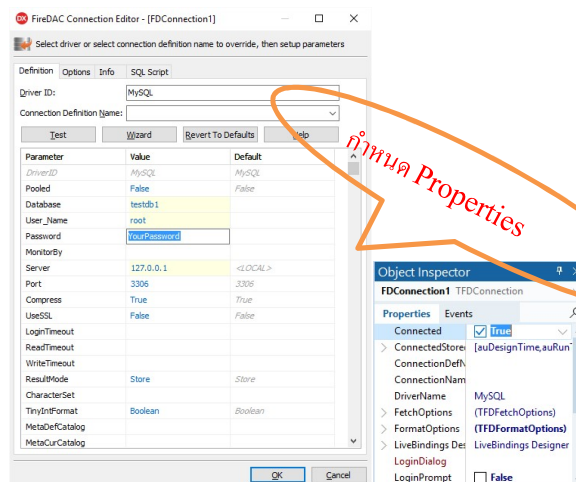
C:\Program Files (x86)\Embarcadero\RAD Studio\20.0\bin

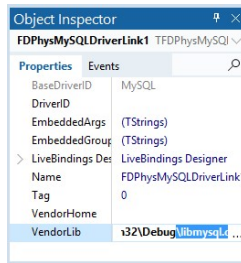


[ProjectFolder\Win32\Debug](#)

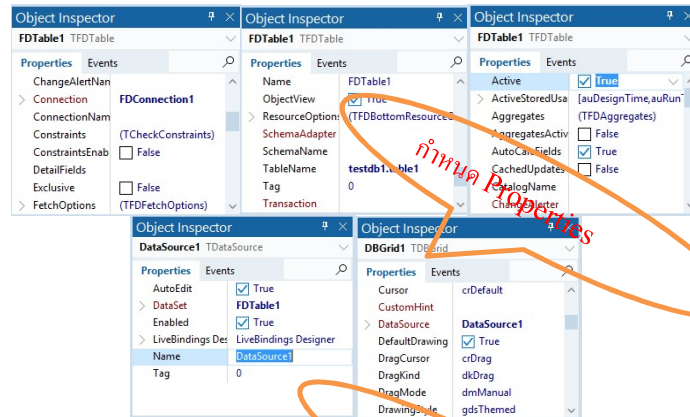


วาง FDConnection, FDPhysMySQLDriverLink, FDTable, DataSource, DBGrid

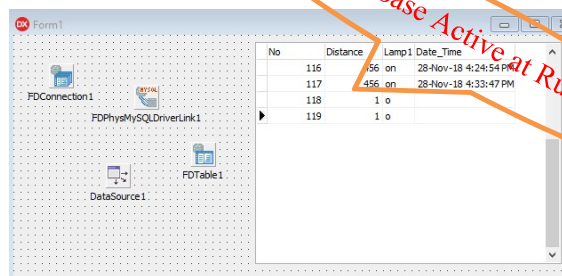




[ProjectFolder\Win32\Debug\libmysql.dll](#)



Result



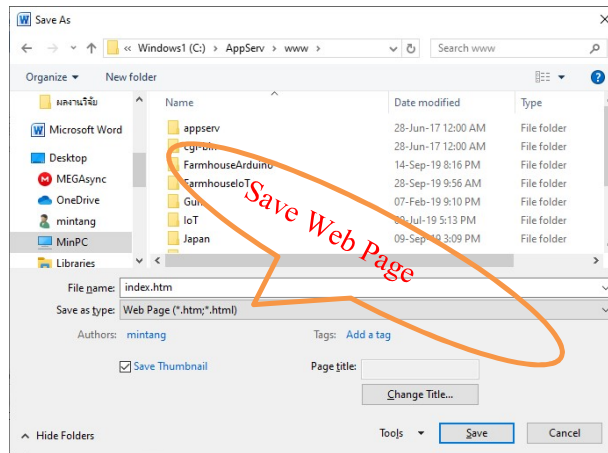
Example

[Code\week12\Ex\\_MySQL\\_1](#)

## 4. ใช้งาน Free Web server, Dashboard และ Database

หลังจากที่ลงโปรแกรม Appserv แล้ว Computer ก็จะสามารถทำหน้าที่เป็น Webserver ได้ ตรวจสอบโดย เข้า <http://127.0.0.1/> หรือ <http://localhost/>

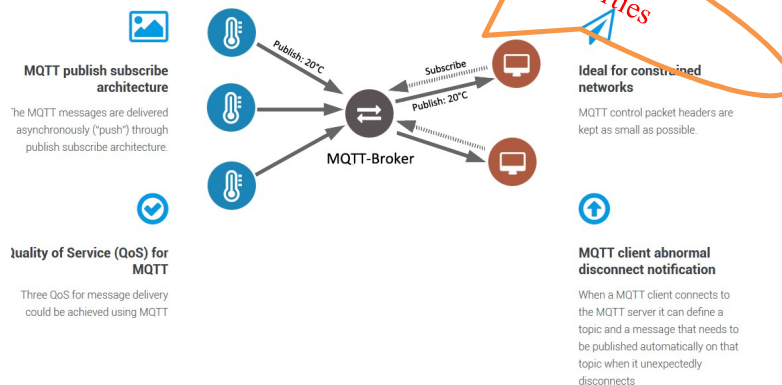
ใช้โปรแกรม Microsoft Word สร้าง webpage โดยให้สร้างเหมือนกับการพิมพ์งานธรรมดา สามารถใส่ ภาพ, ตาราง URL Link และอื่นๆได้ แล้ว Save เป็น Web Page (.htm) ไว้ภายใต้ C:\AppServ\www



ทดสอบโดยเข้า <http://127.0.0.1/index.htm>

## 5. IoT cloud และ Cloud MQTT

หลักการทำงานของ MQTT protocol คือ มี Broker กลางสำหรับรับข้อมูลจาก Publisher มาเก็บไว้ แล้ว Post ข้อมูลดังกล่าวให้กับ Subscriber หรือ Client

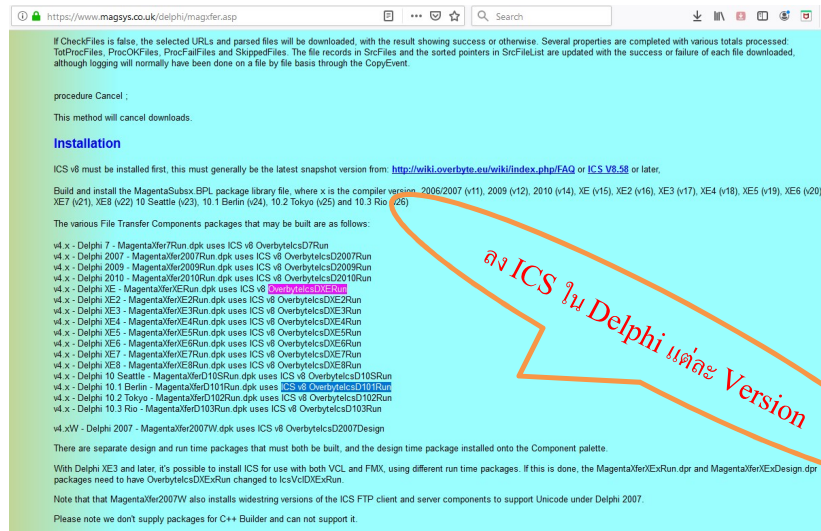


<https://www.esegece.com/websockets/protocols/mqtt>

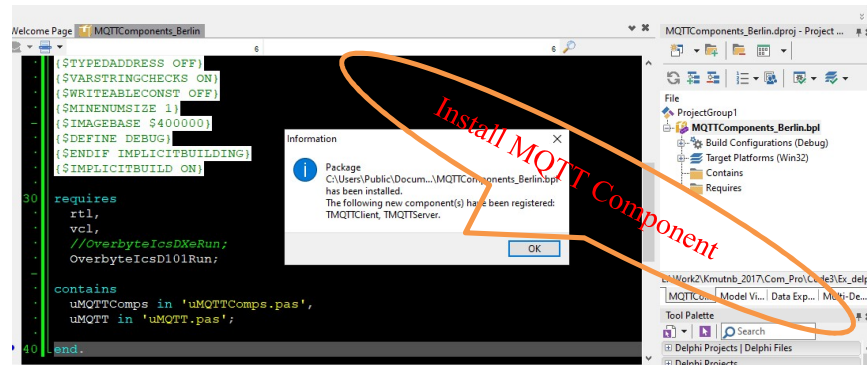
<https://github.com/pjde/delphi-mqtt>

<https://github.com/Virtual-TreeView/Virtual-TreeView>

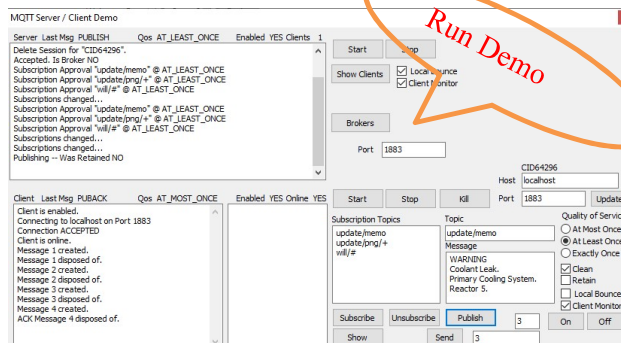
<https://github.com/ashumkin/OverbyteCS/blob/master/Packages/OverbyteCS.DXerun.dproj>



<https://www.magsys.co.uk/delphi/magxfer.asp>



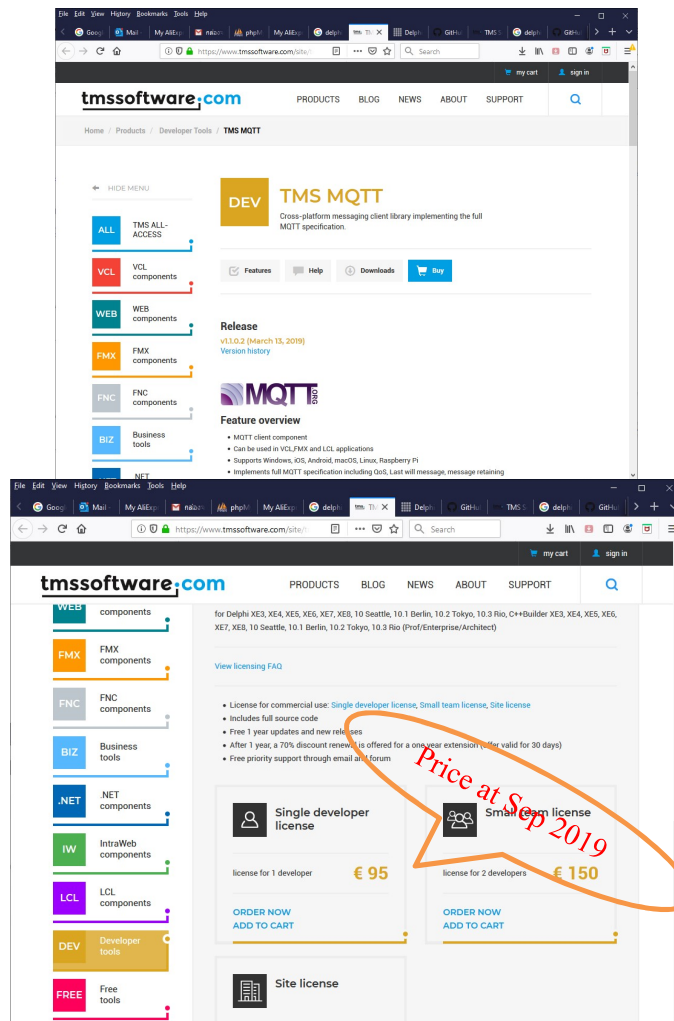
Example



<https://github.com/pjde/delphi-mqtt>

MQTT Component สำเร็จรูป





<https://www.tmssoftware.com/site/tmsmqtt.asp>

## Exercise

- จงเขียน Webpage ที่ให้ข้อมูลเกี่ยวกับตัว นศ.เอง และต้องมี Link ไปยัง Webpage อื่นๆ
- เขียนโปรแกรมบันทึกข้อมูลจากเครื่องจักร (Date, Volt, Amp) ลงใน TextFile และมาแสดงเรียกมาดูใน TStringGrid

## Assignment

- เขียนโปรแกรมบันทึกข้อมูลจากเครื่องจักร (Date, Volt, Amp) ลงใน MySQL และมาแสดงเรียกมาดูใน TDBGrid
- เขียนโปรแกรมรับ - ส่ง MQTT โดยแยกเป็น Broker, Publisher และ Subscriber

## Answer Sheet

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

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

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