|  |
| --- |
| 基本資料 |
| * 姓名：王豪逸 * 標題：B024020014\_Assignment 7(基本) |
| 作業目的與設計 |
| * 目標：感測資料儲存與視覺化 * 電路設計說明(總分1分)：   溫溼度感測器GPIO(2):  接地置於麵包滿最上方，3.3V 置於麵包滿最下方   電路設計照片(總分1分)：  C:\Users\ball\AppData\Local\Microsoft\Windows\INetCache\Content.Word\HW_6.jpg   * 程式邏輯設計(總分1分)：   每分鐘都會去偵測一次溫室度，並將資料連同時間存入mongodb。當大於80%時，會取得前30分鐘的資料製作成圖表，透過FB message API 用粉絲專業傳送訊息與圖表給使用者。   * 影片URL：https://youtu.be/xcto6bwzN9c |
| 心得(總分1分) |
| 有了上次作業的基礎，這次作業的挑戰在生成圖表、時間、FB發送圖片。日期格式上要統一才能夠讓mongodb好搜尋需要的時間點，繪製圖表也需要固定的格式。在Python中datetime較為合適，原本使用time的timestamp但在圖表上較難顯示成時間日期、mongodb有許多參考資料都採用datetime。傳送圖片POST中要附加圖片的二進位檔，無法跟傳送文字的JSON檔使用相同的方法，要使用formdata資料格式修改許多。 |
| 程式碼 |
| 程式碼：  ====基本題===(總分1分)  import Adafruit\_DHT  import datetime, time  import json,requests  from pymongo import MongoClient  import matplotlib.pyplot as plt  import matplotlib.dates as mdates  PLTTIME = datetime.timedelta(minutes=30)  TOKEN = ""  USERID = ""  FILENAME = "./tmp.png"  def main():  sensor = Adafruit\_DHT.DHT11  tempGPIO = 2  client = MongoClient('localhost', 27017)  db = client.test  collect = db.detectdata  print("Start")  try:  sendLock = True  while True:  humi, temp = Adafruit\_DHT.read\_retry(sensor, tempGPIO)  ctime = datetime.datetime.now()  if humi is not None and temp is not None:  print(str(ctime)+" Temp={0:0.1f}\*C Humi={1:0.1f}%".format(temp, humi))  post = {  "time": ctime,  "temp": temp,  "humi": humi  }  collect.insert\_one(post)  if humi >= 80 and sendLock:  sendLock = False  showPLT(collect.find({'time': {'$gte': ctime-PLTTIME, '$lt': ctime}}))  send\_fb\_Image(USERID, FILENAME)  send\_fb\_message(USERID, "Temp={0:0.1f}\*C Humi={1:0.1f}%".format(temp, humi))  else:  sendLock = True  else:  print("Fail to get data")  time.sleep(1)  except KeyboardInterrupt:  ctime = datetime.datetime.now()  showPLT(collect.find({'time': {'$gte': ctime-PLTTIME, '$lt': ctime}}))  def send\_fb\_message(to, message):  post\_message\_url = 'https://graph.facebook.com/v2.6/me/messages?access\_token={token}'  .format(token=TOKEN)  response\_message = json.dumps({"recipient":{"id": to},  "message":{"text":message}})  req = requests.post(post\_message\_url,  headers={"Content-Type": "application/json"},  data=response\_message)  print("[{}] Reply to {}: {}".format(req.status\_code, to, message))  def send\_fb\_Image(to, path):  files = { "filedata" : ('filename.png', open(path, 'rb'), 'image/png')}  data = {  "recipient":'{"id":"' + to + '"}',  "message":'{"attachment":{"type":"image", "payload":{}}}'  }    r = requests.post('https://graph.facebook.com/v2.6/me/messages?access\_token=' + TOKEN, files=files, data=data)  print("[{}] Reply to {}: {}".format(r.status\_code, to, path))  def showPLT(data):  times = []  temps = []  humis = []  for i in data:  times.append(i["time"])  humis.append(i["humi"])  temps.append(i["temp"])  # print(times)  fig=plt.figure(figsize=(20,15))  ax1=fig.add\_subplot(111)  ax1.xaxis.set\_major\_formatter(mdates.DateFormatter('%H:%M'))  ax1.axis([times[0], times[-1], 0, 100])  ax1.plot(times, temps,'ro-', linewidth= 5)  ax2=ax1.twinx()  ax2.axis([times[0], times[-1], 0, 100])  ax1.plot(times, humis,'bo-', linewidth= 5)  fig.savefig(FILENAME)  if \_\_name\_\_=="\_\_main\_\_":  main() |