

Chap X 範例

作者：蔣文得

飲食與健康管理 APP 範例

現代人因生活步調快速，許多人會選擇食用外食，這導致台灣超過四成的人口有肥胖問題(資料來源:風傳媒報導)，在攝取不均且運動量不足的情況下，罹患糖尿病和心血管疾病的風險迅速飆升。

根據 衛服部國民健康屬 統計，全國約有 200 多萬名糖尿病的病友，且每年以 25,000 名的速度持續增加，使得糖尿病被列為 2016 年國人十大死因之第五名，心血管疾病則被列為十大死因之第二名，可見此現象的嚴重性。

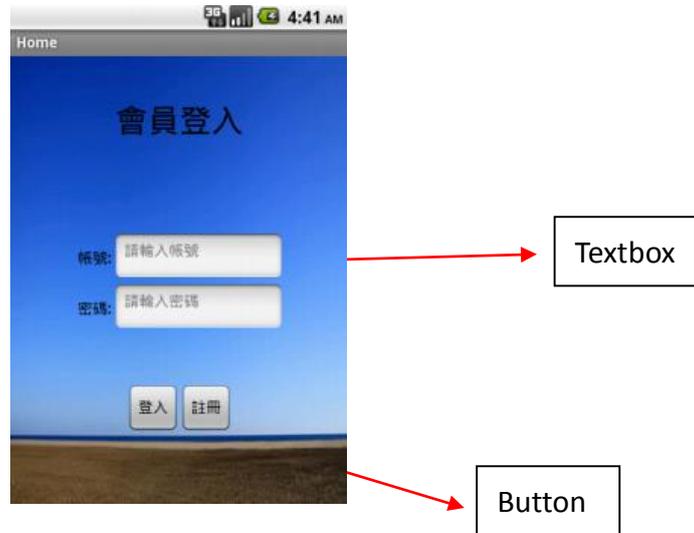
根據上述案例，我們可以透過 App Inventor 2 開發一支 App 協助現代人記錄飲食狀況及運動狀況，提供使用者輸入目前的健康狀況，並且可透過掃描 QR Code 將食物營養含量輸入手機，推算使用者每日建議的營養攝取與建議使用者該多攝取那些食物?是否有攝取不足或超標?也可提供相關健康資訊讓使用者做為參考(如: 正確運動姿勢、正確飲食觀念等...)，鞏固使用者的健康。

此案例中我們將使用到程式語言中的 變數、陣列、事件、循序概念、選擇結構及迴圈，使用到的 App Inventor 2 元件為 計步器(Pedometer)、圖像(Image)、對話框(Notifier)、按鈕(Button)、標籤(Label)、清單選擇器(ListPicker)、介面配置(Layout)等...元件，一步一步帶著學生進行開發。總結以上我們可將 app 的需求整理成列表如下：

需求名稱	需求說明	對應課程	
需求 1	記錄運動量	以計步器記錄運動量	Pedometer
需求 2	記錄飲食	可將飲食以掃描 QR Code 的方式記錄下來	DatePicker FirebaseDB BarcodeScanner
需求 3	顯示飲食記錄	依照選取日期顯示飲食記錄	Label DatePicker ListView
需求 4	顯示每日統計	依照選取日期顯示當日的營養統計與運動量，並顯示評價	DatePicker Label Button

需求 5	畫面警示	在清除所有資料前警告 使用始確認是否清除資料	Notifier
需求 6	選取日期	可讓使用者選取日期	DatePicker
需求 7	檢視記錄	可供使用者檢視飲食記錄	ListView FirebaseDB
需求 8	換頁	可切換頁面	Control
需求 9	取得當天日期	取得當天日期以供 App 執 行記錄	Clock
需求 10	輸入食物	使用 QR Code 來輸入當天 吃的食物	BarcodeScanner
其餘需求	按鈕、變數	程式中所需要的資 料可暫時存放於變數	Button Variables

用 App Inventor 2 設計出的程式畫面如下



登入畫面

```

when btn_register .Click
do
  call Notifier1 .ShowAlert
  notice " 施工中 "

initialize global member_info to create empty list

initialize global account to ""

when btn_login .Click
do
  set global account to txt_account .Text
  set global password to txt_password .Text
  call FirebaseDB1 .GetValue
  tag join "tb_member/"
  get global account
  "/password"
  valueIfTagNotThere ""

initialize global password to ""

when FirebaseDB1 .GetValue
tag value
do
  if
  get value != "" and get global password = get value
  then
    call TinyDB1 .StoreValue
    tag "Account"
    valueToStore get global account
    open another screen screenName "Home"
  else
    call Notifier1 .ShowAlert
    notice " 登入失敗，請檢察您的帳號和密碼是否正確。 "
  
```

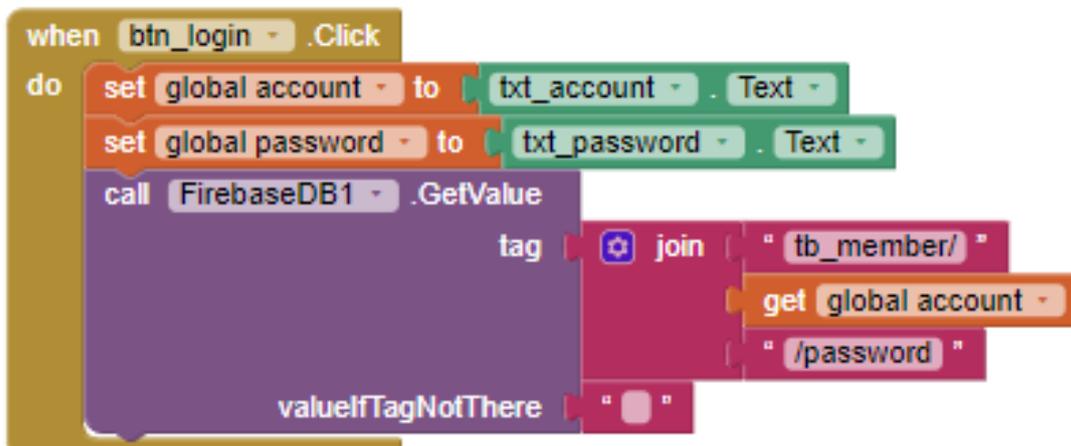
登入畫面程式區塊

宣告變數是寫程式缺一不可的步驟，如同數學裡的變數可以用來解決許多題目一樣，程式裡的變數也有著異曲同工之妙。在程式設計中，我們時常需要記錄某些資料，這些資料有可能是文字，也有可能是數字，我們把這些資料記錄在記憶體某個位址中，並給他一個名稱，這就是變數。變數大致上分成兩種，一種是全域變數，一種是區域變數。全域變數就是宣告在所有函式之外，使所有的函式都可以共用這個變數；相反的，區域變數就是宣告在某些函式之中，使這個變數只能被這個函式所使用，並不能被其他函式所共同使用。

下面是整個主畫面的程式都可以使用的全域變數。在後面的 blocks 中可看到這些變數的使用方式。



登入按鈕(btn_login)負責將登入資訊送進資料庫程序裡面檢查，把輸入的帳號和密碼存入 account 和 password 兩個變數中，並且送入 Firebase 的檢查程序來驗證登入。

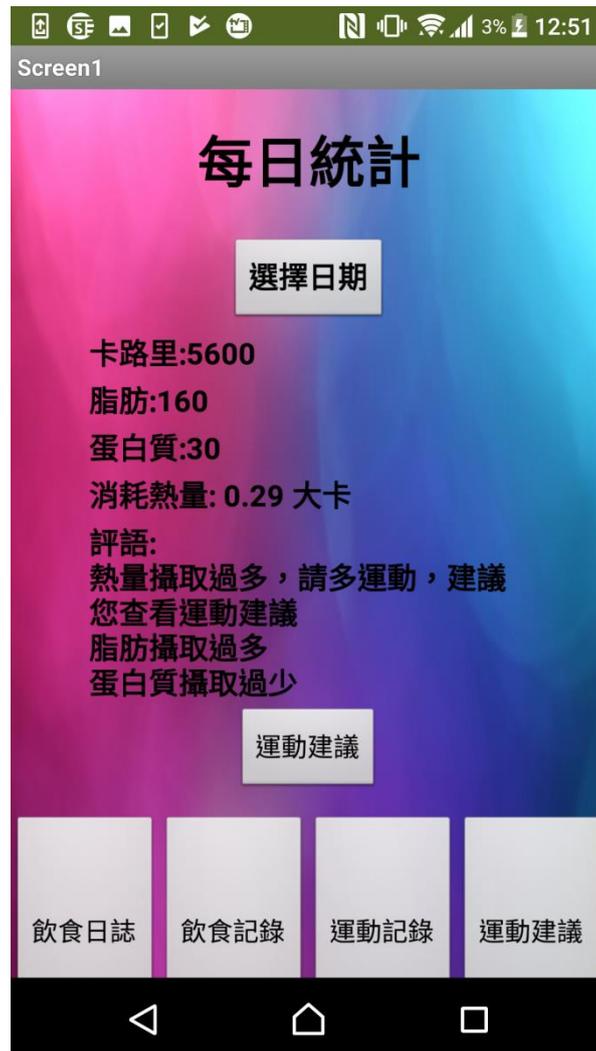


下面的 FirebaseDB1 這個元件負責檢查資料庫中有沒有這個會員，若有的話則登入成功，進入主選單。

```

when FirebaseDB1 .GetValue
  tag value
do
  if
    get value ≠ "" and get global password = get value
  then
    call TinyDB1 .StoreValue
      tag "Account"
      valueToStore get global account
    open another screen screenName "Home"
  else
    call Notifier1 .ShowAlert
      notice "登入失敗，請檢察您的帳號和密碼是否正確。"

```



每日統計畫面



每日統計選擇日期
畫面

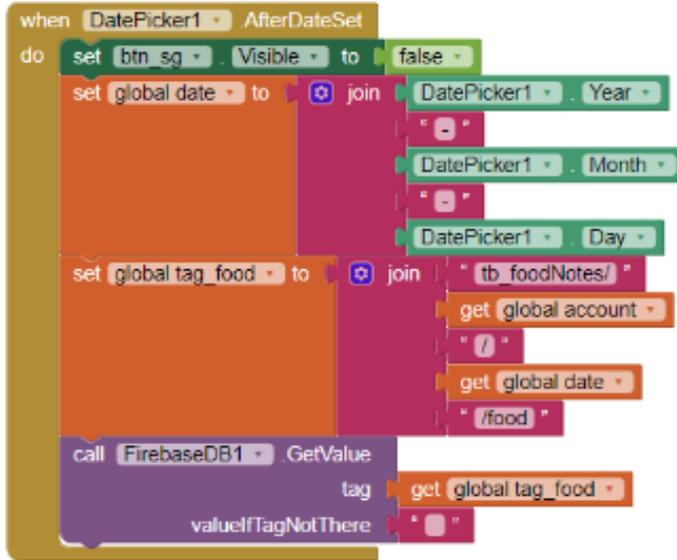
下圖是每日統計(Home)的初始化區塊，首先從 TinyDB 中讀取已經登入的使用者帳號來驗證身分，接下來將必要的 Label 參數做初始化。

```

when Home.Initialize
do
  set global account to call TinyDB1.GetValue
  tag "Account"
  valueIfTagNotThere ""
  set lbl_cal.Text to "卡路里: 0"
  set lbl_fat.Text to "脂肪: 0"
  set lbl_protein.Text to "蛋白質: 0"
  set lbl_info.Text to "評語:\n無"
  set btn_sg.Visible to false

```

下圖是每日統計畫面中的 DatePicker，讓使用者選擇想觀看的日期，進入 Firebase 中搜尋該日期的飲食統計資料並且顯示。



下方為本頁必要的變數做初始化。



這是 Firebase 的處理區塊，將所選的日期與帳號帶入 Firebase 中並取得當日飲食統計資訊，若沒有取得資訊則表示當日並沒有記錄。

```

when FirebaseDB1 .GetValue
  tag value
do
  if get tag = get global tag_food
  then
    if get value = ""
    then
      call Notifier1 .ShowAlert
        notice "當日無任何記錄"
      set lbl_cal .Text to "卡路里: 0"
      set lbl_fat .Text to "脂肪: 0"
      set lbl_protein .Text to "蛋白質: 0"
      set lbl_heat .Text to "消耗熱量: 0"
      set lbl_info .Text to "評語: 無"
    else
      set global tag_info to join
        "tb_foodNotes/"
        get global account
        "/"
        get global date
        "/info"
      call FirebaseDB1 .GetValue
        tag get global tag_info
        valueIfTagNotThere ""
  
```

以下是畫面下方四個按鈕的設定，負責跳轉到相應的頁面。

```

when btn_Exercise .Click
do
  open another screen screenName "ExerciseRecord"

when btn_FoodNotes .Click
do
  open another screen screenName "FoodNotes"

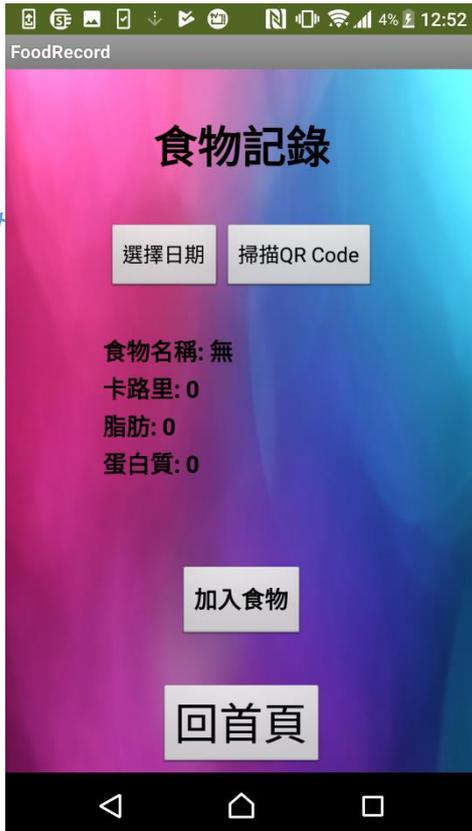
when btn_Suggest .Click
do
  open another screen screenName "ExerciseSuggest"

when btn_sg .Click
do
  open another screen screenName "ExerciseSuggest"
  
```

以下是給予使用者每日統計評語的邏輯，使用數學類別的比較大小來分別判斷熱量、脂肪與蛋白質是否超標。

The image shows a sequence of logic blocks in APP Inventor 2. It starts with a 'set lbl_info . Text to' block containing the text '評語:\n'. This is followed by three conditional blocks. The first is an 'if' block with the condition 'get global cal < 1600'. The 'then' branch sets the text to '熱量攝取不足，請補充熱量\n'. The 'else if' block has the condition 'get global cal > 2300', and its 'then' branch sets the text to '熱量攝取過多，請多運動，建議您查看運動建議\n' and also sets 'btn_sg . Visible' to 'true'. The 'else' branch sets the text to '熱量攝取正常\n'. The second conditional block is an 'if' block with the condition 'get global fat > 40', with a 'then' branch setting the text to '脂肪攝取過多\n'. The 'else if' block has the condition 'get global cal < 15', with a 'then' branch setting the text to '脂肪攝取過少\n'. The 'else' branch sets the text to '脂肪攝取正常\n'. The final conditional block is an 'if' block with the condition 'get global protein < 50', with a 'then' branch setting the text to '蛋白質攝取過少\n' and an 'else' branch setting the text to '蛋白質攝取正常\n'.

```
set lbl_info . Text to "評語:\n"  
if (get global cal < 1600)  
  then set lbl_info . Text to join(lbl_info . Text, "熱量攝取不足，請補充熱量\n")  
  else if (get global cal > 2300)  
    then set lbl_info . Text to join(lbl_info . Text, "熱量攝取過多，請多運動，建議您查看運動建議\n")  
        set btn_sg . Visible to true  
    else set lbl_info . Text to join(lbl_info . Text, "熱量攝取正常\n")  
if (get global fat > 40)  
  then set lbl_info . Text to join(lbl_info . Text, "脂肪攝取過多\n")  
  else if (get global cal < 15)  
    then set lbl_info . Text to join(lbl_info . Text, "脂肪攝取過少\n")  
    else set lbl_info . Text to join(lbl_info . Text, "脂肪攝取正常\n")  
if (get global protein < 50)  
  then set lbl_info . Text to join(lbl_info . Text, "蛋白質攝取過少\n")  
  else set lbl_info . Text to join(lbl_info . Text, "蛋白質攝取正常\n")
```



DatePicker

食物記錄初始畫面



掃描食物上貼的 QR code 新增食物



成功掃描到食物的畫面



成功新增食物的畫面

以下是食物記錄的初始化區塊，從 TinyDB 讀取使用者的帳號，初始化 Label 的文字，並使用 Clock 將預設要記錄的時間設定為今天。

```

when FoodRecord.Initialize
do
  set global account to call TinyDB1.GetValue
  tag "Account"
  valueIfTagNotThere ""
  set lbl_name.Text to "食物名稱: 無"
  set lbl_cal.Text to "卡路里: 0"
  set lbl_fat.Text to "脂肪: 0"
  set lbl_protein.Text to "蛋白質: 0"
  call DatePicker1.SetDateToDisplay
  year call Clock1.Year
  instant call Clock1.Now
  month call Clock1.Month
  instant call Clock1.Now
  day call Clock1.DayOfMonth
  instant call Clock1.Now
  set global date to join call Clock1.Year
  instant call Clock1.Now
  "-"
  call Clock1.Month
  instant call Clock1.Now
  "-"
  call Clock1.DayOfMonth
  instant call Clock1.Now

```

也可以藉由 DatePicker 手動選擇時間

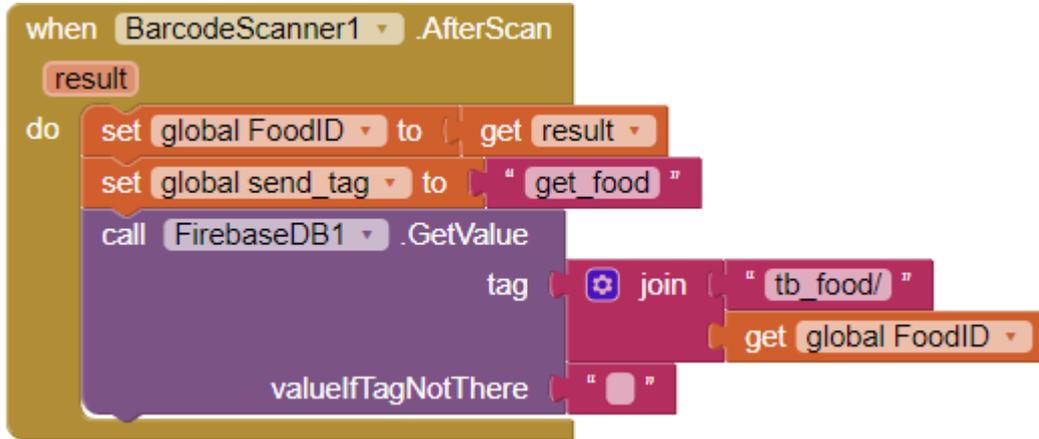
```

when DatePicker1.AfterDateSet
do
  set global date to join DatePicker1.Year
  "-"
  DatePicker1.Month
  "-"
  DatePicker1.Day
  set global food_tag to join "tb_foodNotes/"
  get global account
  "/"
  get global date
  "/food"

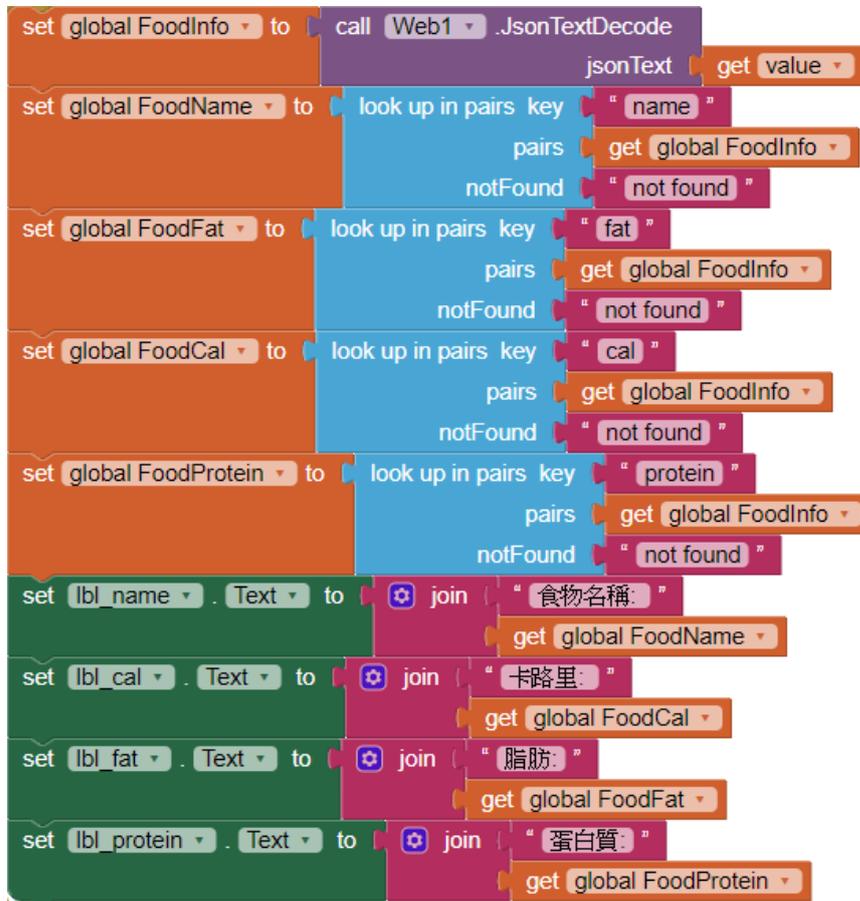
```

當按下掃描 QR Code 時，程式會啟動 BarcodeScanner，並且將掃描到的結果傳送至資料庫。

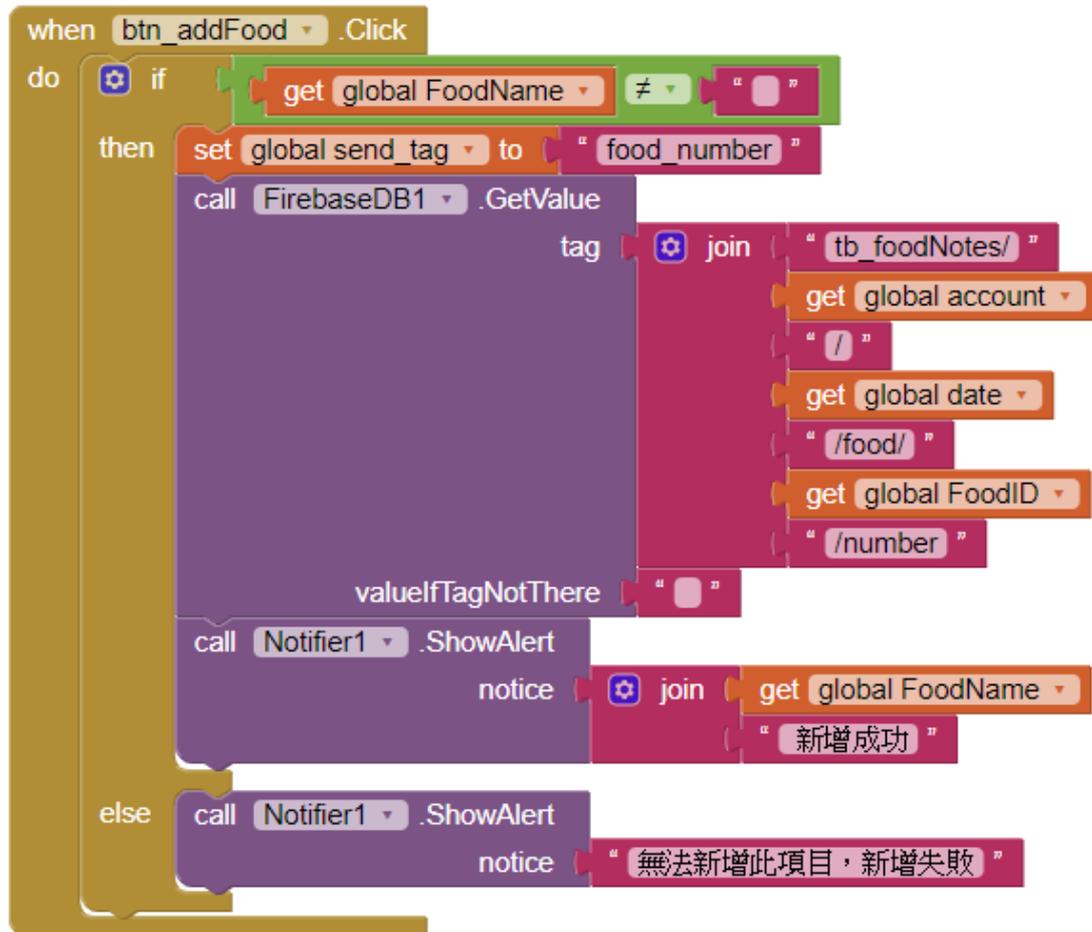
以食物為例，若想要以 QR Code 掃描食物並且讓程式識別出該食物，必須先將該食物新增至 Firebase 資料庫中，再將該食物的編號以 QR Code 產生器產生 QR Code，貼在食物上即可供本 App 掃描。



一旦 Firebase 識別出該食物，那麼就會顯示該食物的圖片和營養數據等...資訊。值得注意的是，Firebase 回傳的資料是 json 型態，必須使用 App Inventor 2 提供的 Web 元件來進行解碼並存入 list 元件中，再使用“look up in pairs”功能來讀取。



程式成功識別食物後，就可以按加入食物按鈕來將該食物加入本日的飲食清單，同樣的，程式會將食物資訊依照使用者帳號和日期，以食物編號的方式儲存再 Firebase 資料庫中。





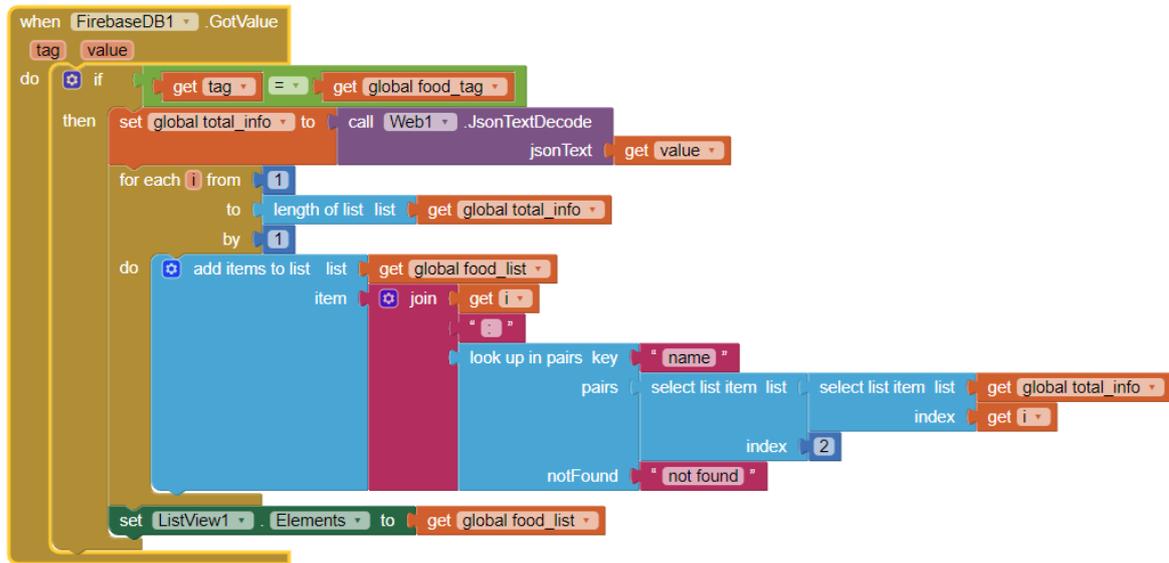
飲食記錄選擇日期畫面



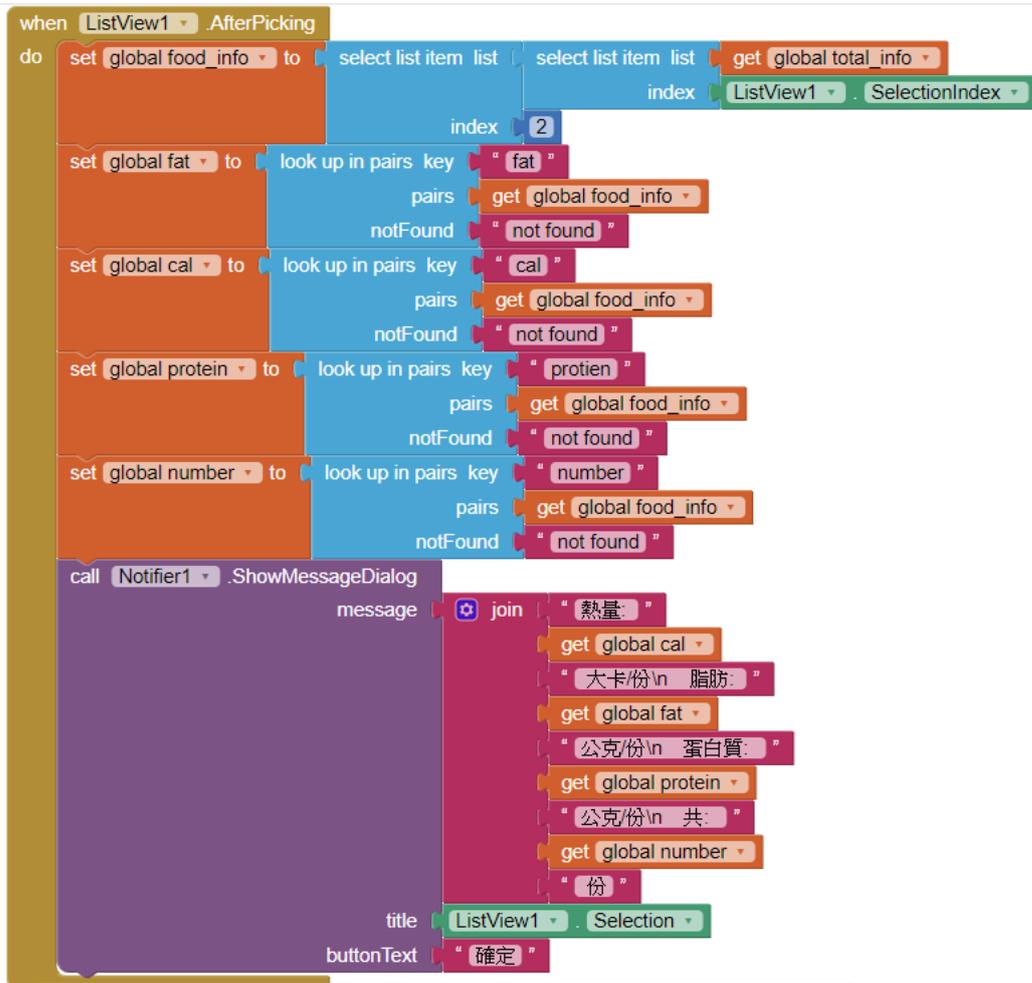
Notifier

飲食詳細資訊

使用 DatePicker 選擇完日期後，程式會到 Firebase 中搜索並傳回該帳號在該日期的飲食記錄存入 list 中。



當使用者點選某個食物時，程式將會以 Notifier 元件來顯示該食物的營養資訊和當天食用的分量。





運動記錄畫面

當點選開始運動時，程式會啟動計步器和計時器。

```

when btn_start .Click
do
  set global account to call TinyDB1 .GetValue
  tag "Account"
  valueIfTagNotThere ""
  set btn_pause .Enabled to true
  set btn_resume .Enabled to false
  set btn_start .Enabled to false
  set btn_stop .Enabled to true
  set global km to 0
  set global hours to 0
  set global seconds to 0
  set global total_distance to 0
  set global total_walk to 0
  call Pedometer1 .Start
  call Notifier1 .ShowAlert
  notice "開始記錄"
  set Clock1 .TimerEnabled to true
  set lbl_distance .Text to "總距離: 0 公里"
  set lbl_walk .Text to "行走步數: 0 步"
  set lbl_speed .Text to "平均時速: 0 km/h"
  set lbl_time .Text to "時間: 00:00:00"

```

計步器的程式區塊如下。

```

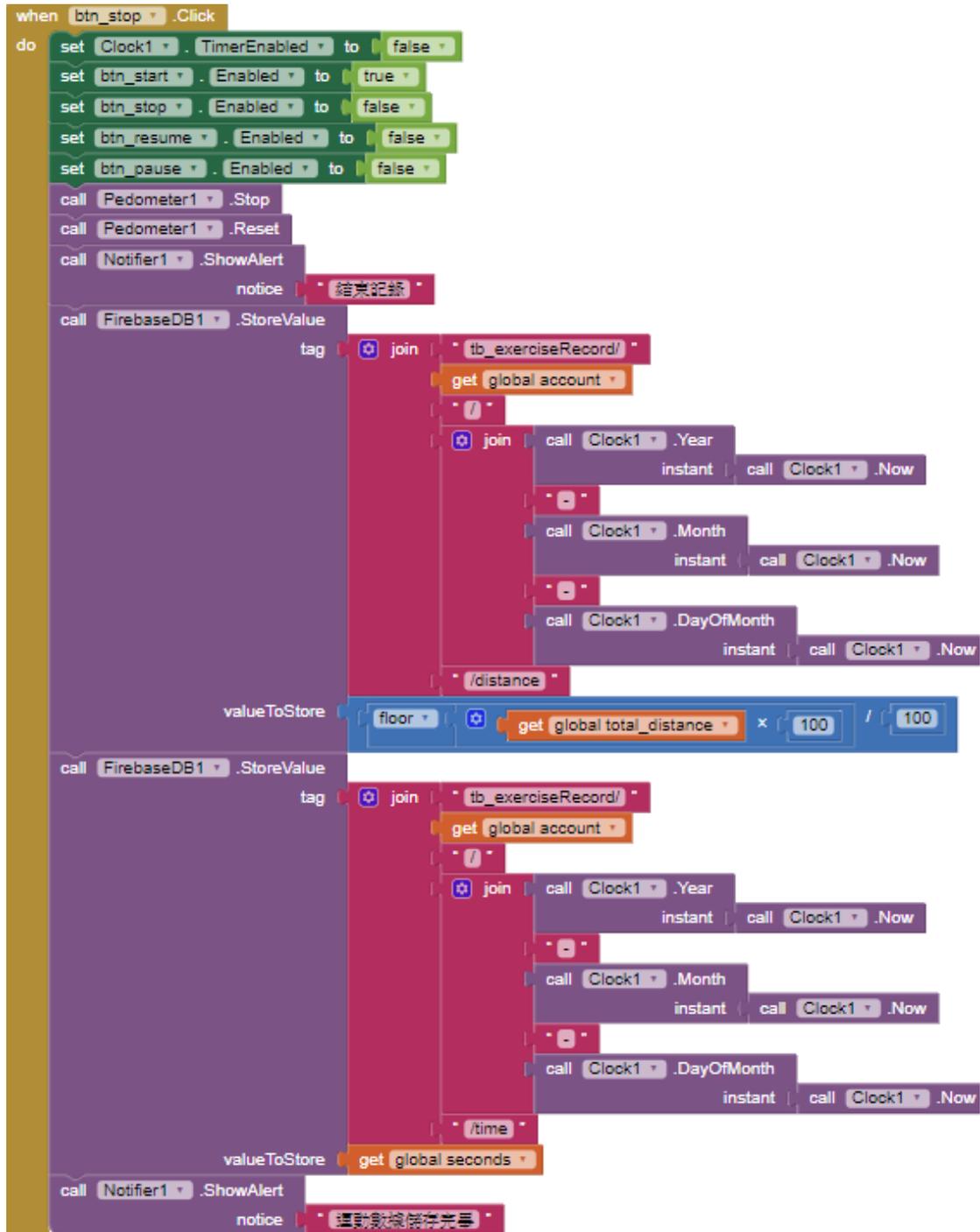
when Pedometer1 .WalkStep
  walkSteps distance
do
  set global total_distance to Pedometer1 .Distance
  set global total_walk to Pedometer1 .WalkSteps
  set lbl_distance .Text to join "總距離:"
  floor
  get global total_distance
  / 10
  × 10
  / 1000
  "公里"
  set lbl_walk .Text to join "行走步數:"
  get global total_walk
  "步"

```

運動過程中程式會計算您的平均時速，公式如下。



按下停止按鈕則運動終止，將運動記錄回傳 Firebase。



下方是計時器的運作邏輯。

```

when Clock1.Timer
do
  set global seconds to (get global seconds + 1)
  set global km to (floor (get global total_distance / 10) * 10 / 1000)
  set global hours to (get global seconds / 3600)
  if (get global hours != 0)
  then
    set lbl_speed.Text to (join ("平均速度: "
      (floor (get global total_distance / 1000) / get global hours * 100 / 100)
      " km/h"))
  set global h to (floor (get global seconds / 3600))
  set global m to (modulo of (floor (get global seconds / 60) + 60))
  set global s to (modulo of (get global seconds + 60))
  if (get global h < 10)
  then
    set global h to (join ("0"
      get global h))
  if (get global m < 10)
  then
    set global m to (join ("0"
      get global m))
  if (get global s < 10)
  then
    set global s to (join ("0"
      get global s))
  set lbl_time.Text to (join ("時間: "
    get global h
    ":"
    get global m
    ":"
    get global s))

```

下方是暫停和恢復按鈕的運作邏輯。

```

when btn_pause.Click
do
  set Clock1.TimerEnabled to false
  set btn_resume.Enabled to true
  set btn_pause.Enabled to false
  call Pedometer1.Pause
  call Notifier1.ShowAlert
  notice "暫停記錄"

when btn_resume.Click
do
  set Clock1.TimerEnabled to true
  set btn_resume.Enabled to false
  set btn_pause.Enabled to true
  call Pedometer1.Resume
  call Notifier1.ShowAlert
  notice "繼續記錄"

```



運動建議畫面

運動建議畫面的邏輯如下，此處使用 `WebView` 元件來嵌入 Youtube 影片。

```

when btn_hard .Click
do
  set btn_easy .Enabled to true
  set btn_hard .Enabled to false
  set lbl_type .Text to "類型: 有氧高強度間歇"
  set lbl_burn .Text to "消耗: 約 500 kal/hr (iFit 愛瘦身 提供)"
  set lbl_describe .Text to "說明: 適合想燃燒更多脂肪的人。"
  call WebView1 .GoToUrl
  url "https://www.youtube.com/embed/xAUV1oMdiWw"

```