

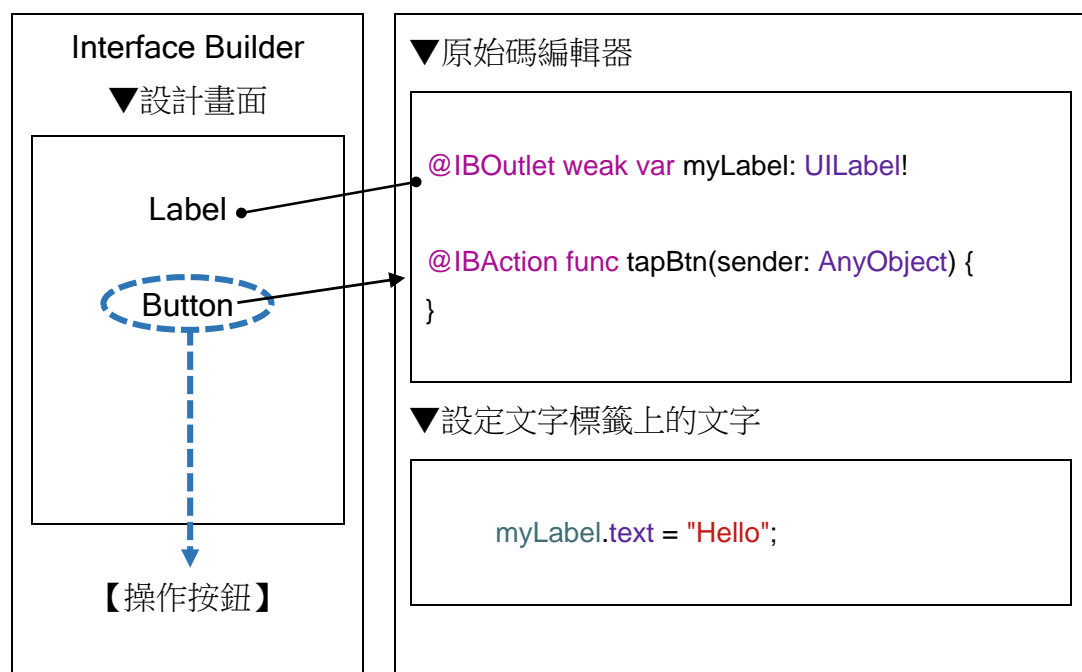
CHAPTER 5-2

UIButton：當按下按鈕後可執行指定動作

點選按鈕顯示文字

學習概念：

1. 首先用 IB 建立【文字標籤】及【操作按鈕】。
2. 接著將【文字標籤】、【操作按鈕】與【程式碼連結】。
3. 最後在實作檔中實作相關程式，在處理畫面載入後所觸發的事件，也就是在撰寫按下【操作按鈕】後，可以顯示【文字標籤】上內容的程式。

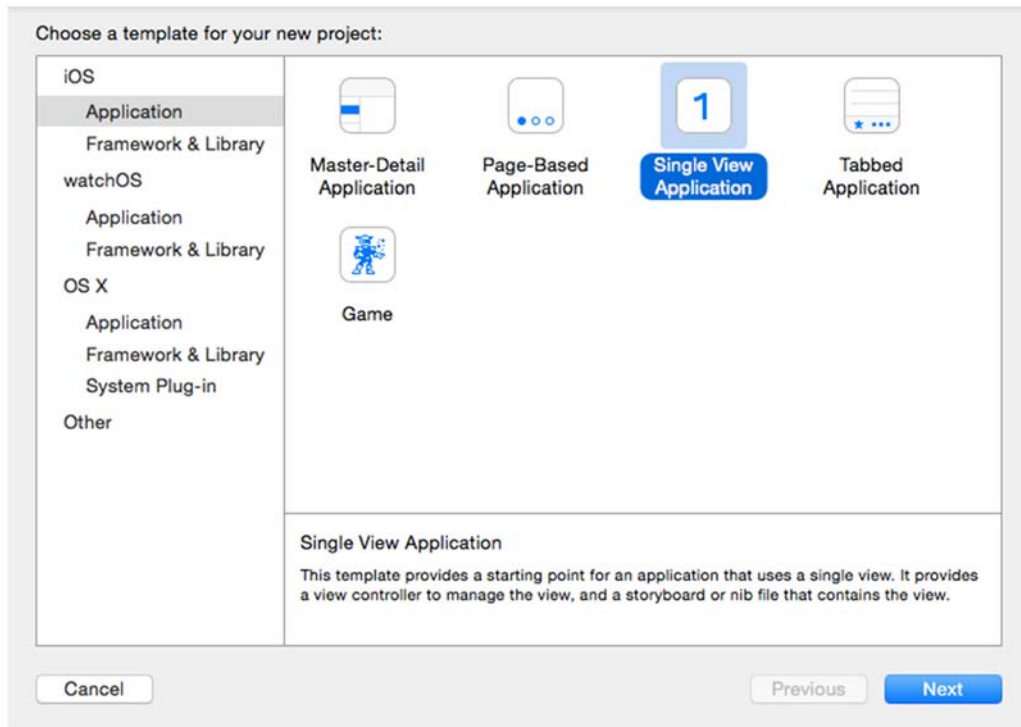


【執行結果】

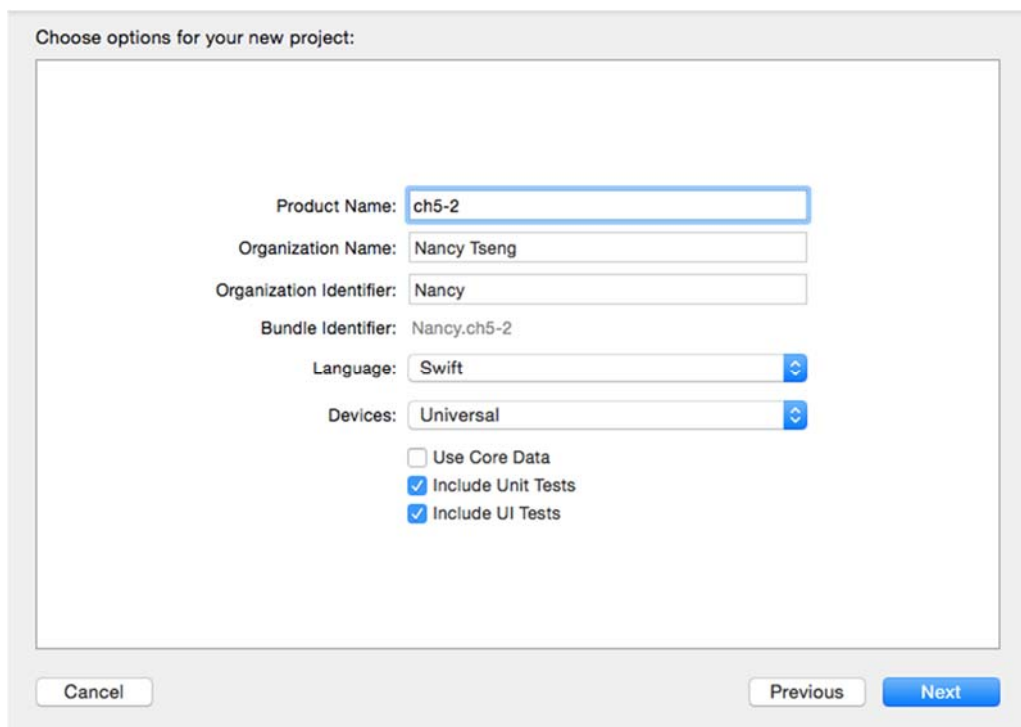
當 App 執行後，設定在【文字標籤】上的文字，透過使用者的執行【操作按鈕】，將顯示在【myLabel】。

Step.1

開啟 **xcode** 時會出現的畫面，點選 **iOS** 下的【**Application**】，接著右視窗選擇【**Single View Application**】，點選【**Next**】選項後進入設定的基本視窗。

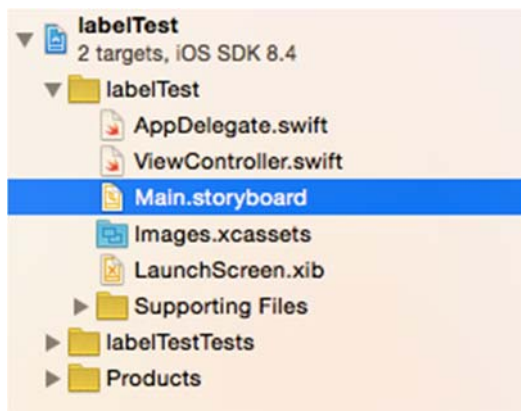


檔名及名稱設定，請將【**Product Name**】設定為 **ch5-2**

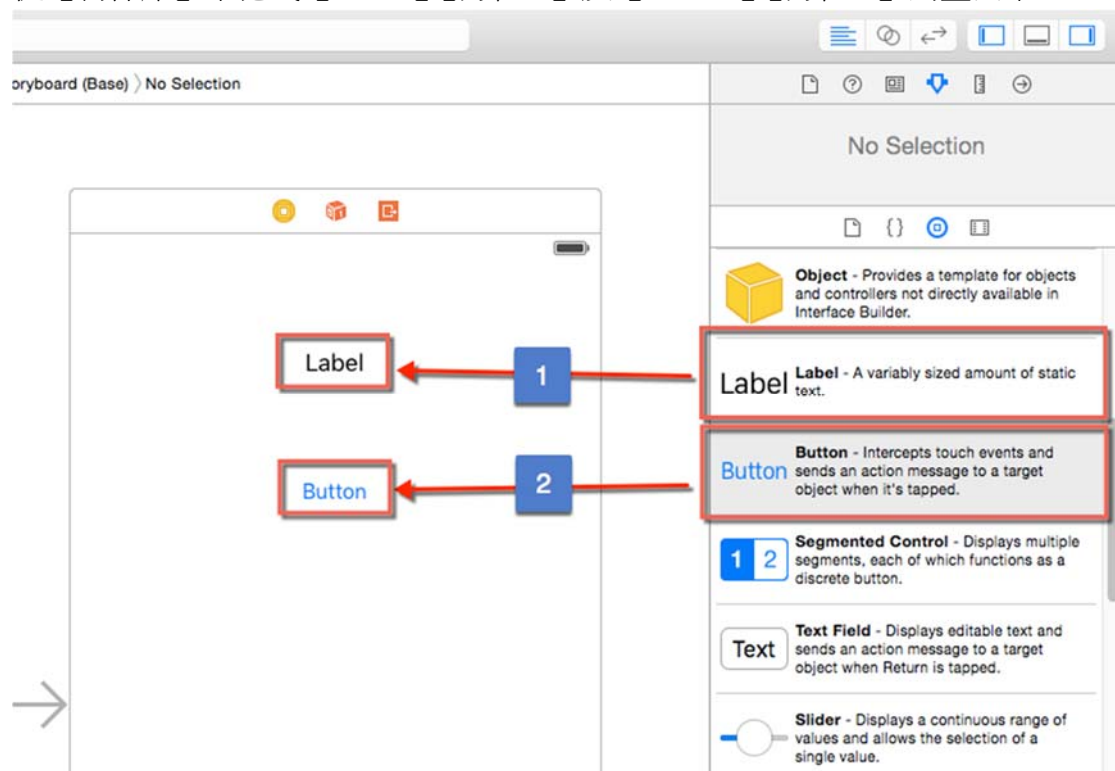


Step.2

選取【Main.storyboard】



本章操作已選擇【iPhone 4.7-inch】 操作頁面。(詳見 5-1 屬性設定小技巧)
從【物件庫】中拖曳【Label】【方框 1】及【Button】【方框 2】到畫面中。



Step.3.

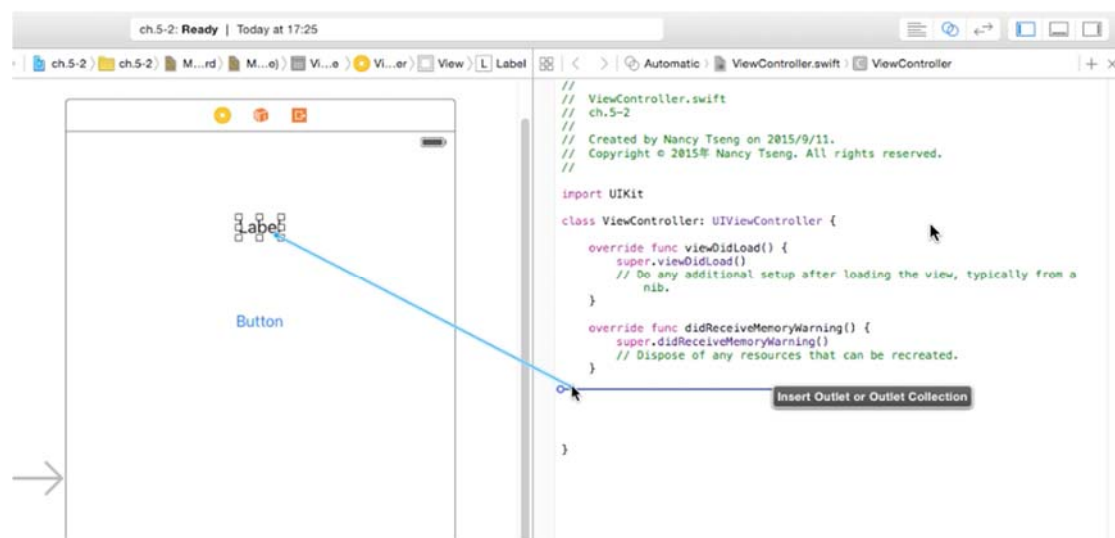
接著點選右上方工具列視窗【輔助編碼器】，就是雙圈符號【2】，進行程式碼編輯。



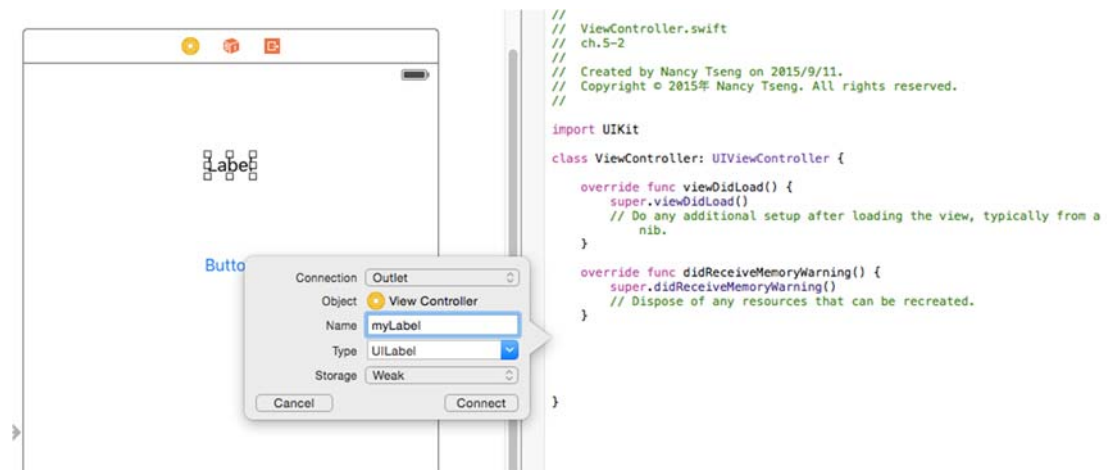
(詳見 5-1 屬性設定小技巧)

Step.4

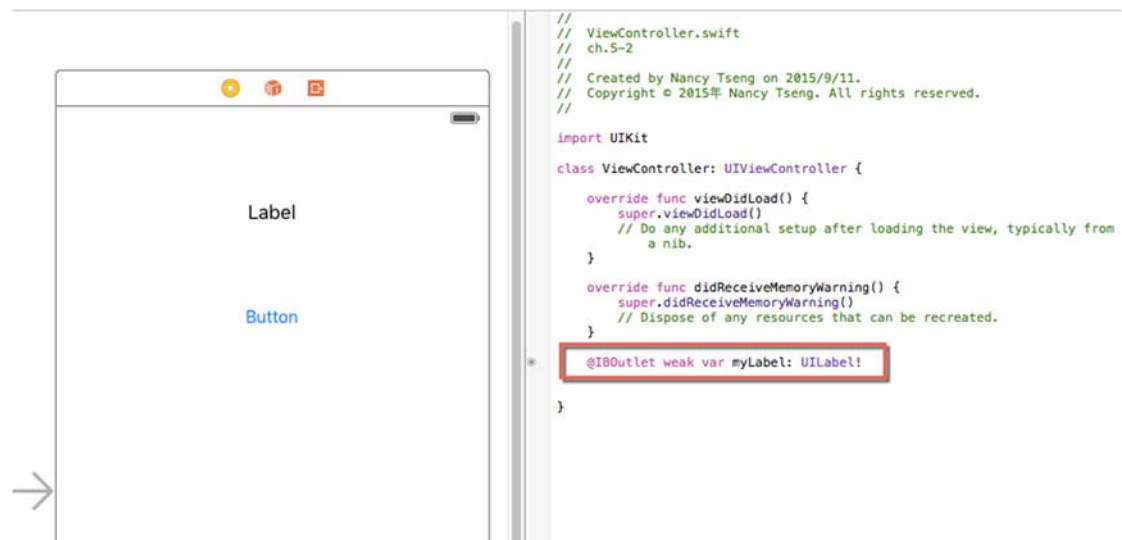
- (1) 將【文字標籤】和【名稱連結】。
- (2) 在 IB 視窗上點選【文字標籤】，接著按住【control】鍵，用滑鼠拖曳【文字標籤】到右邊視窗與【程式碼】連結。



- (3) 此時出現彈跳視窗，我們命【文字標籤】名稱為【myLabel】後，按【Connect】按鈕。

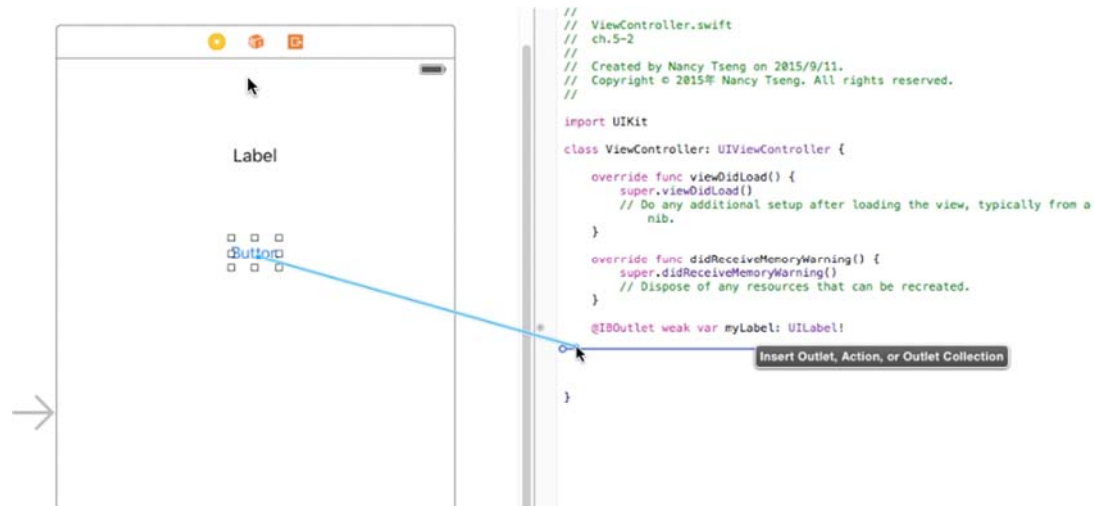


(4) 透過連結即產生紅框處的程式碼，控制 IB 建立的元件。

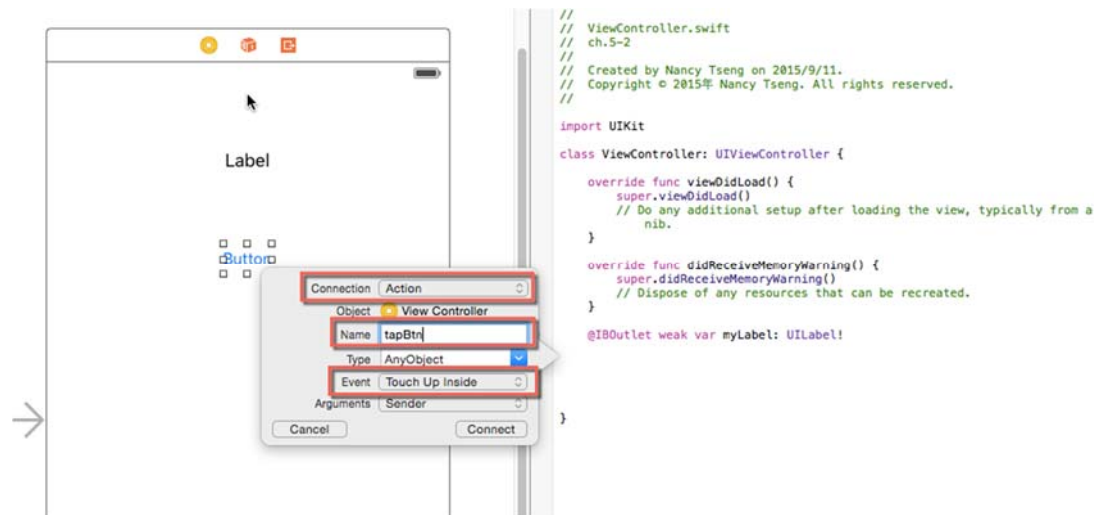


Step.5

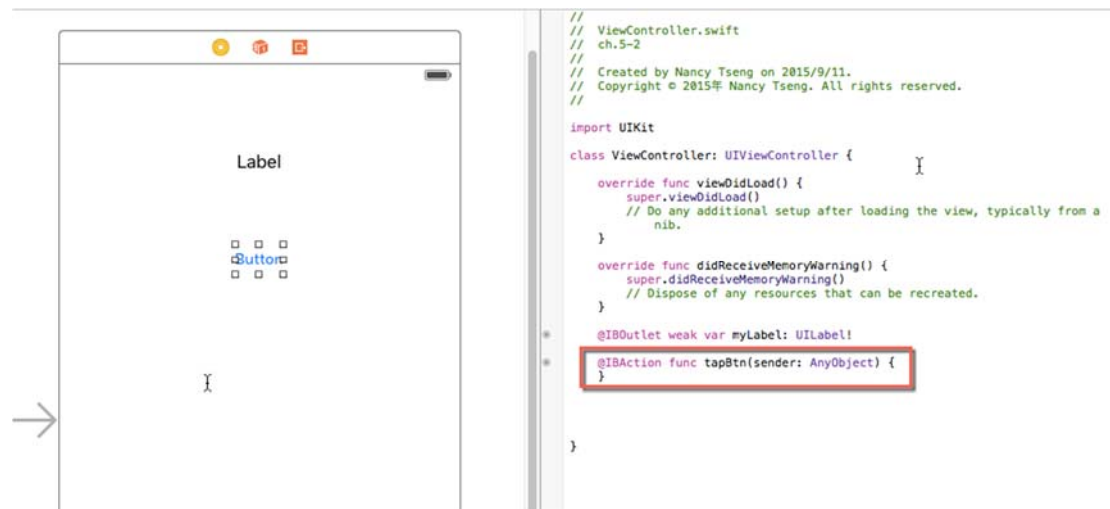
- (1) 將【操作按鈕】和【程式碼】連結。
- (2) 在 IB 視窗上點選【操作按鈕】，接著按住【control】鍵，用滑鼠拖曳【操作按鈕】到右邊視窗與【程式碼】連結。



- (3) 在「Connection」欄位點選【Action】、「Name」欄位設定名稱【tapBtu】以及在「Event」欄位點選【Touch Up Inside】後，按【Connect】按鈕。

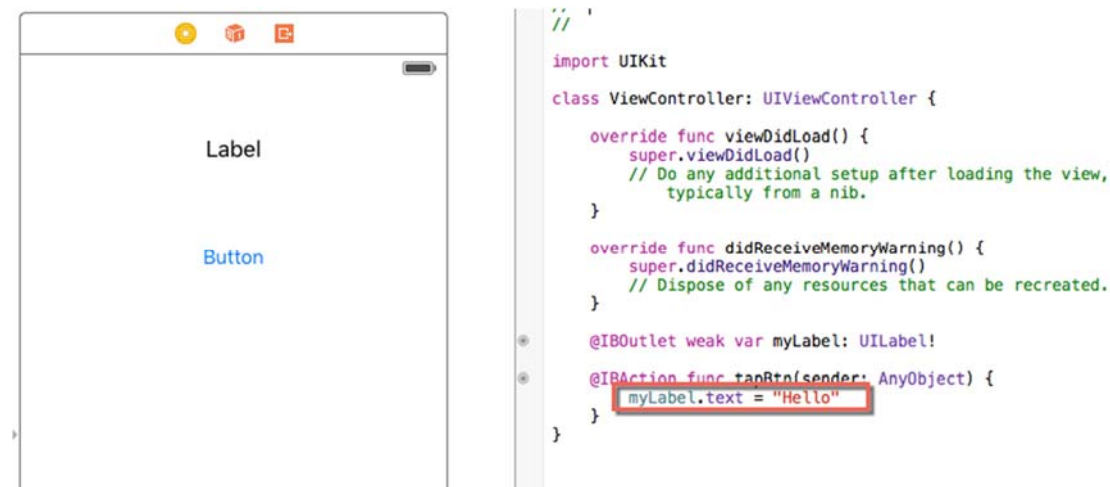


- (4) 透過連結即產生紅框處的程式碼，控制 IB 建立的元件。



(5) 於 IBAction tapBtn 中加入程式碼。

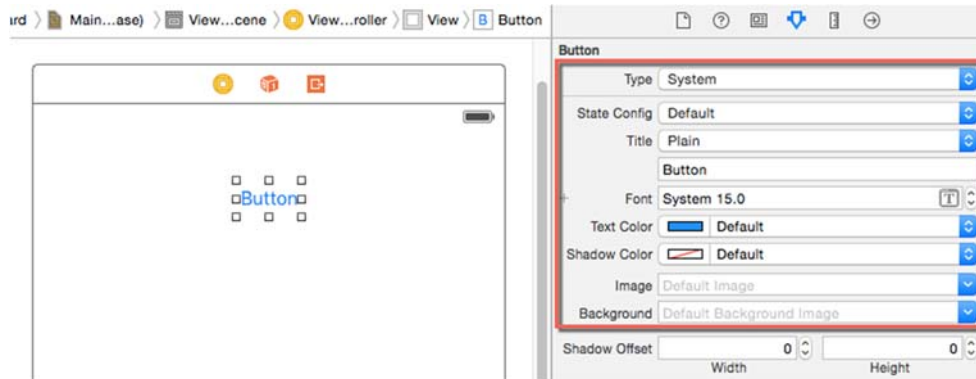
`myLabel.text = "Hello"`





屬性設定小技巧

關於【Button】常用屬性說明：



選 項		功 能 說 明
1.	Typ	按鈕的種類
2.	Title	按鈕的文字
3.	Font	按鈕上文字大小
4.	Image	按鈕上所顯示的圖片
5.	Background	按鈕的背景圖片
6.	Text Color	文字顏色

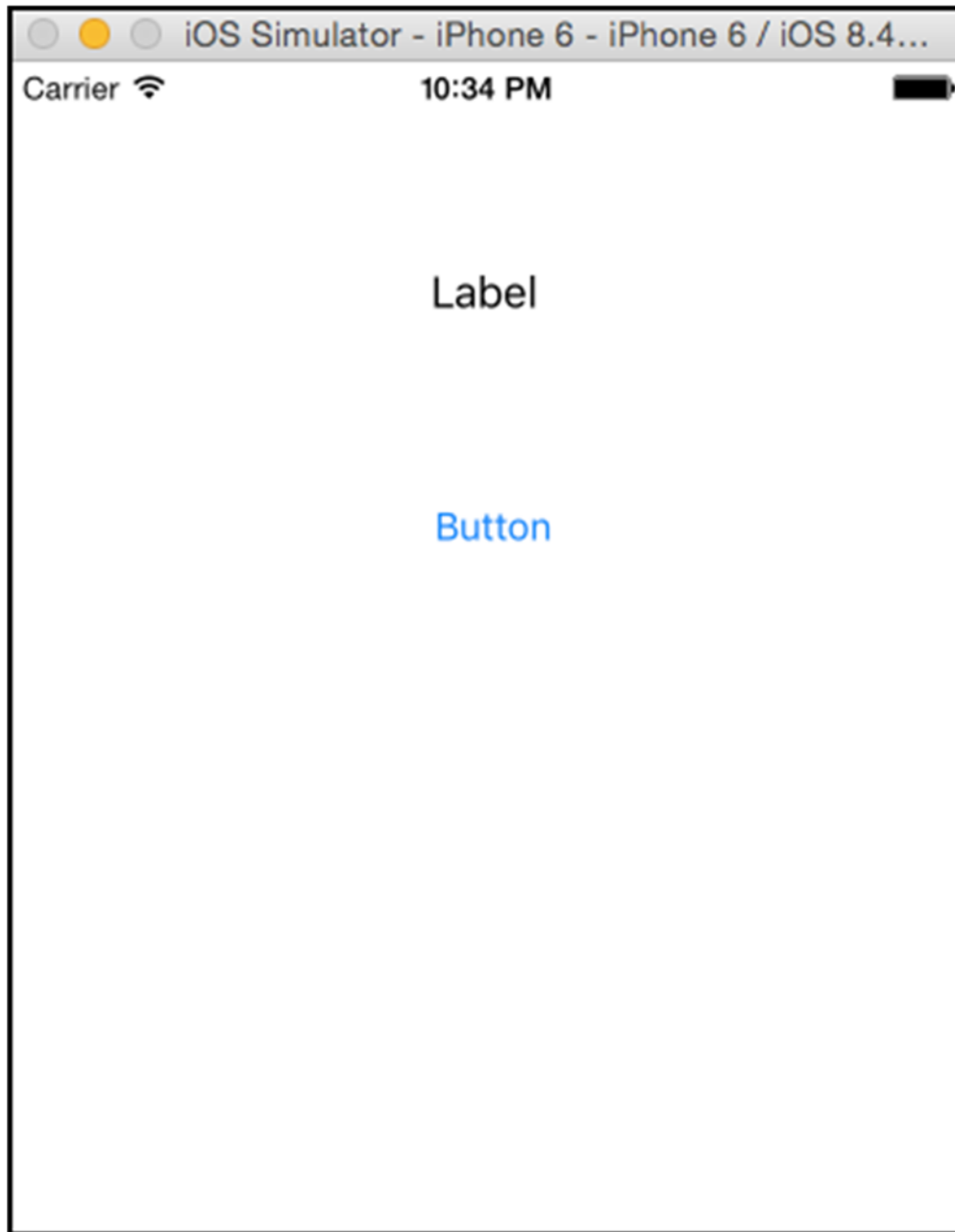
Step.6.

在上方工具列按下【執行鍵▶】(Build and then run the current scheme)，啟動模擬器執行程式。

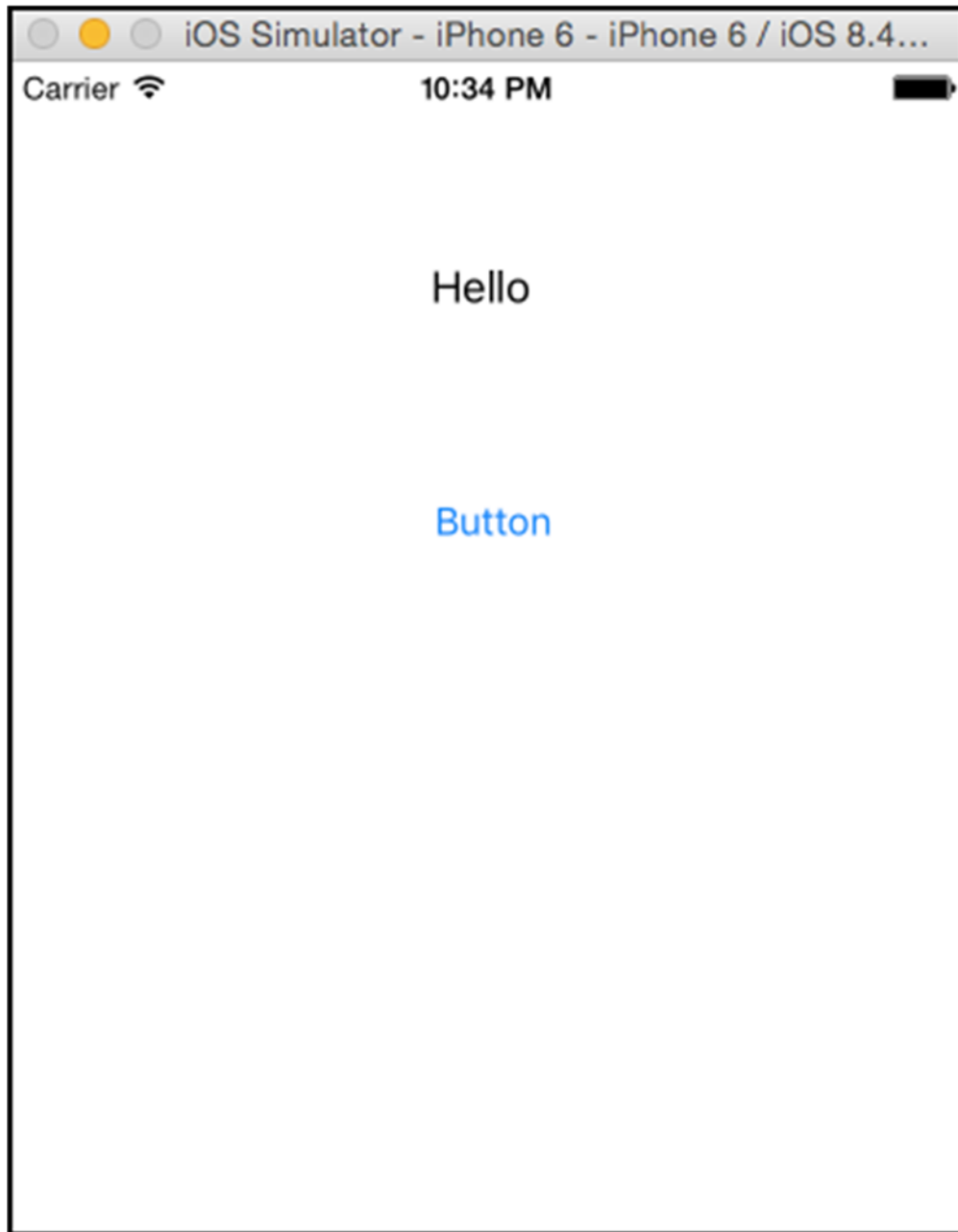


Step.7

當 App 啟動後在顯示畫面時，將【myLabel】的顯示文字設定為「Label」。



點選【操作按鈕】(Button) App 啟動後在顯示畫面，將【myLabel】的顯示文字為「Hello」。



(小編在這裡設定為 iPhone6 畫面，供各位讀者們參考)

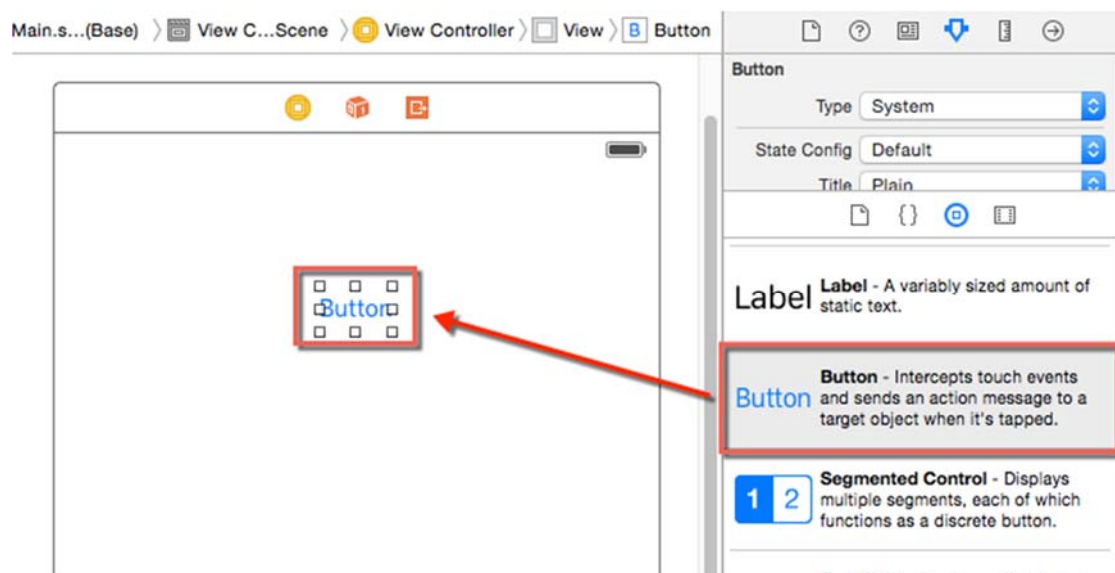
透過上面簡單的【Button】操作按鈕介紹，以下還有更進階用法，供讀者參考練習。

活用【Button】進階一

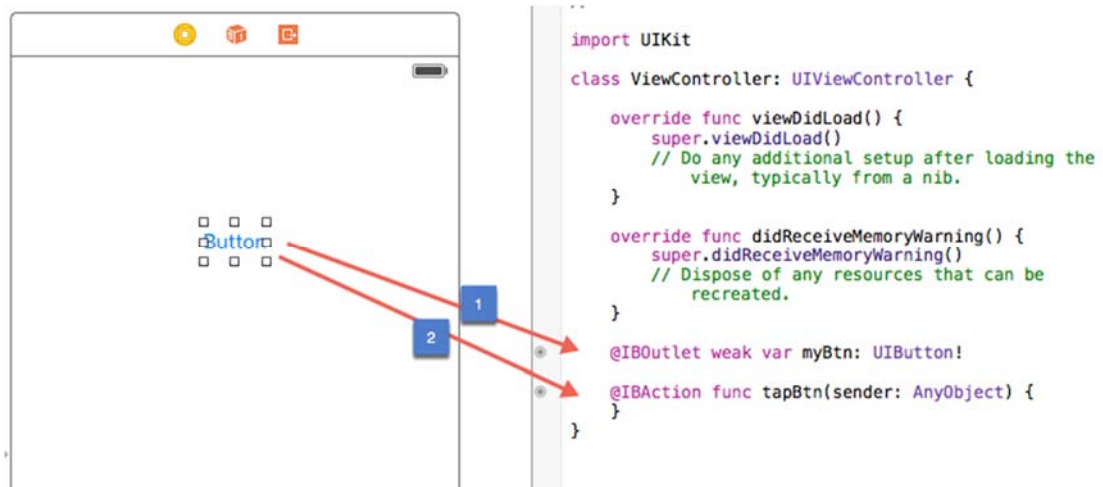
學習概念：

使用【BUTTON】程式執行時，動態顯示【BUTTON】名稱【按一下】，當按下【BUTTON】後，立即變更【BUTTON】名稱為【已按過】。

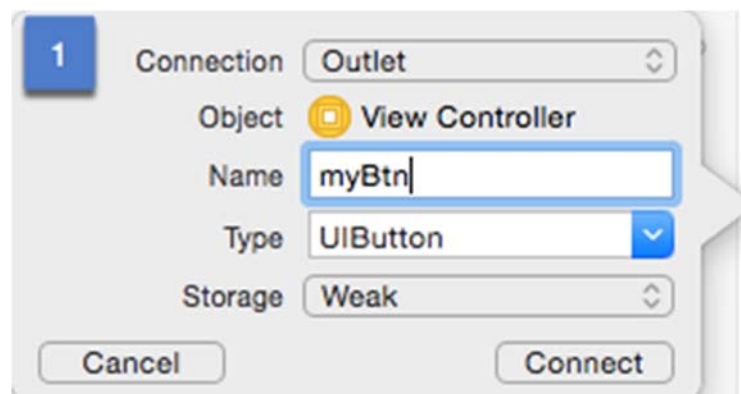
1. 將【操作按鈕】建立在畫面中。



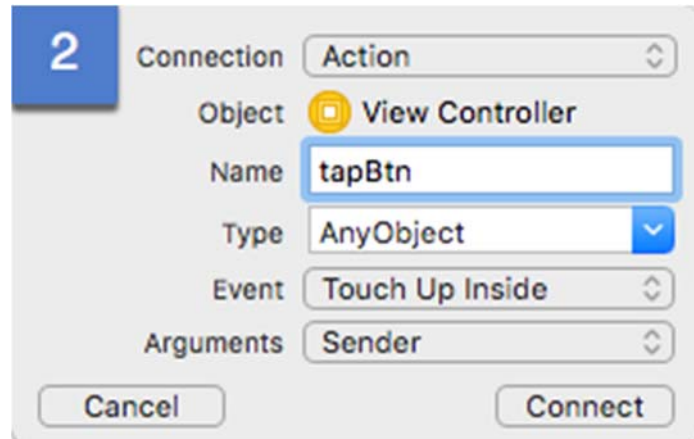
2. 在 IB 視窗上點選【操作按鈕】，接著按住【control】鍵，用滑鼠拖曳【操作按鈕】到右邊視窗與【程式碼】連結。



3. 在「Connection」欄位點選【Outlet】、「Name」欄位設定名稱【myBtn】以及在「Type」欄位點選【UIButton】後，按【Connect】按鈕。



4. 在「Connection」欄位點選【Action】、「Name」欄位設定名稱【tapBtn】，「Type」欄位點選【AnyObject】以及在「Event」欄位點選【Touch Up Inside】後，按【Connect】按鈕。



最後，設定 swift 的程式碼。

```
myBtn.setTitle("已按過", forState: UIControlState.Normal)
```

```
@IBOutlet weak var myBtn: UIButton!  
  
@IBAction func tapBtn(sender: AnyObject) {  
    myBtn.setTitle("已按過", forState:  
        UIControlState.Normal)  
}
```



屬性設定小技巧

Swift 語言教學：

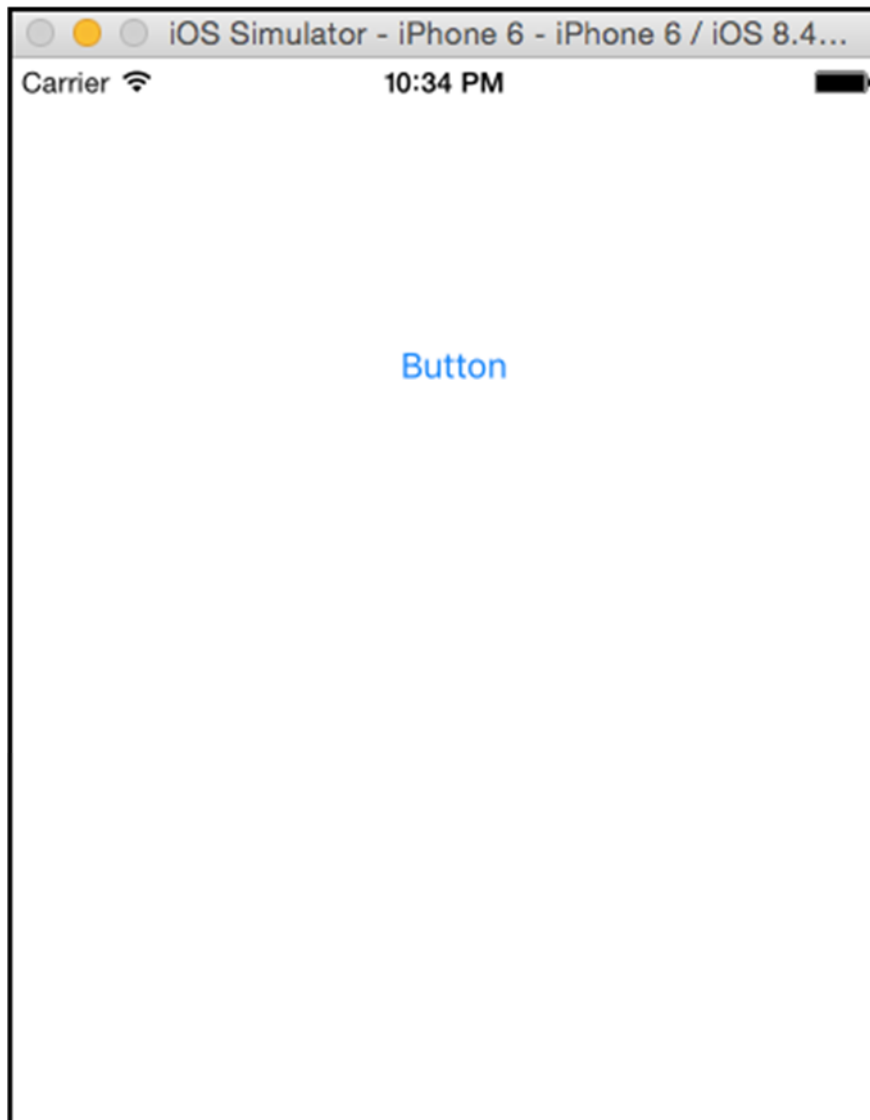
【設定操作按鈕標題】

透過操作按鈕 **setTitle** 屬性，顯示內容設定：

```
操作按鈕.setTitle("myTitle", forState: UIControlState.Normal);
```

【forState】為按鈕狀態設定，而【UIControlState.Normal】則是一般正常狀態。

當 App 啟動後在顯示畫面時，透過【操作按鈕】的操作顯示文字為「已按過」。



（請按一下畫面中的【Button】，端看有會起什麼樣的變化。）



恭喜你！自己是不是對建置程式多了份信心，喝杯水我們繼續下一個練習吧！
（小編在這裡設定為 iPhone6 畫面，供各位讀者們參考囉）

自我練習

實作：

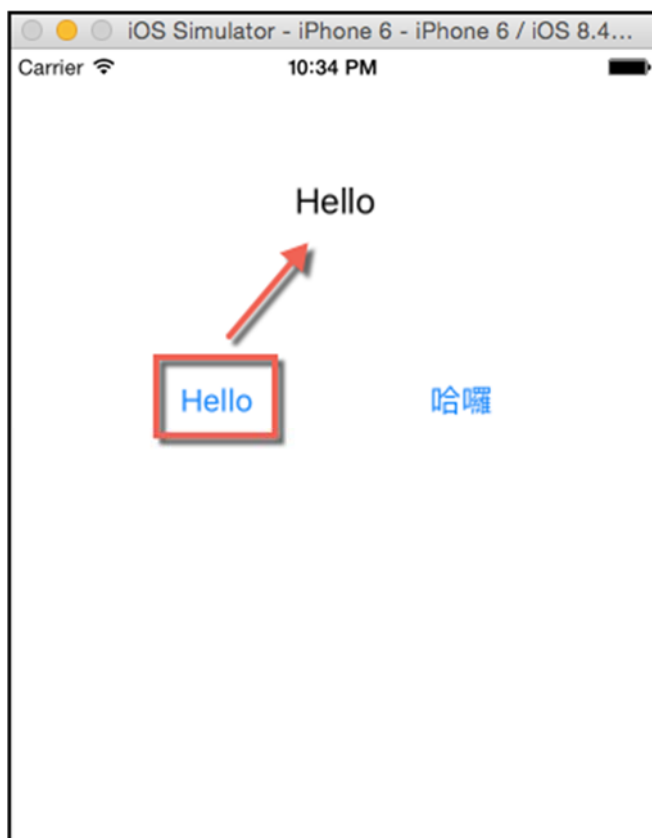
使用兩個【BUTTON】程式執行時，動態顯示【BUTTON】名稱為「Hello」以及「哈囉」，當按下其中之一的【BUTTON】後，即變更【BUTTON】為指定【Label】名稱。

當 App 啟動後在顯示畫面時，透過【操作按鈕】的操作顯示文字為「Hello」或「哈囉」。

（請按一下畫面中的「Hello」【操作按鈕】，端看有會起什麼樣的變化。）



選項「英文字體」的【操作按鈕】，而【文字標籤】隨之顯示為「Hello」。



選項「中文字體」的【操作按鈕】，而【文字標籤】隨之顯示為「哈囉」。

