

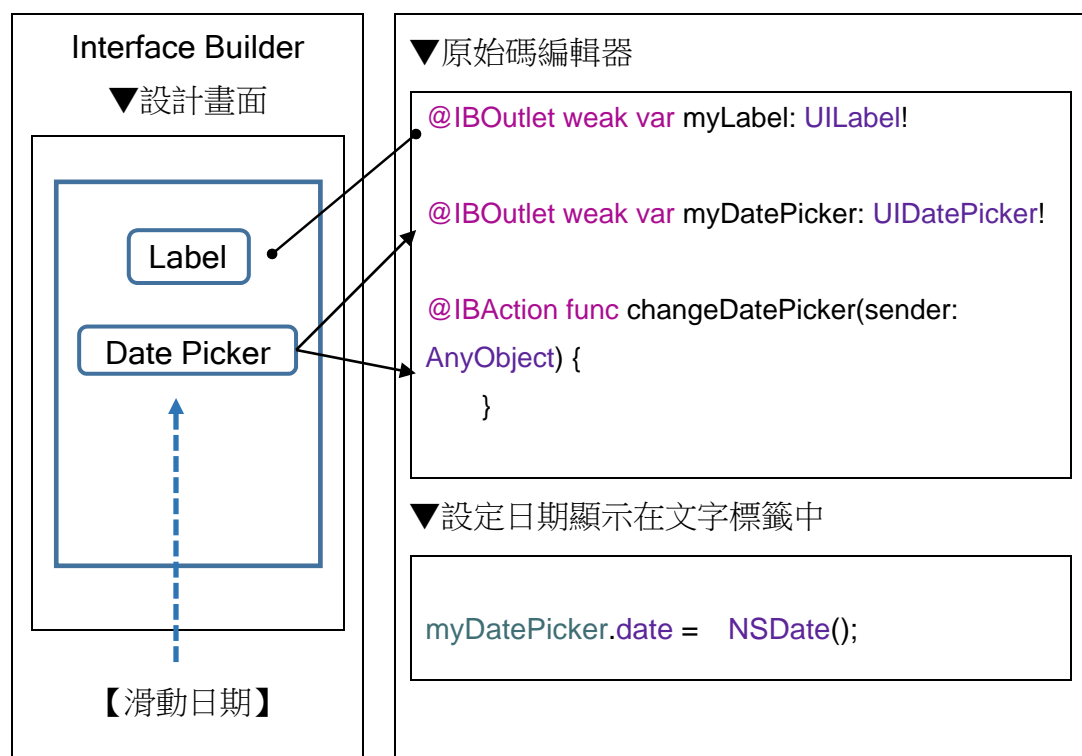
CHAPTER 5-7

UIDatePicker：讓日期時間顯示在文字標籤上

文字標籤裡顯示日期及時間

學習概念：

1. 首先用 IB 建立【文字標籤】及【日期時間方塊】。
2. 將【文字標籤】及【日期時間方塊】與【程式碼】連結。
3. 最後在實作檔中相關程式，於處理載入後所觸發的事件，也就是撰寫利用【日期時間方塊】顯示在【文字標籤】上的程式。

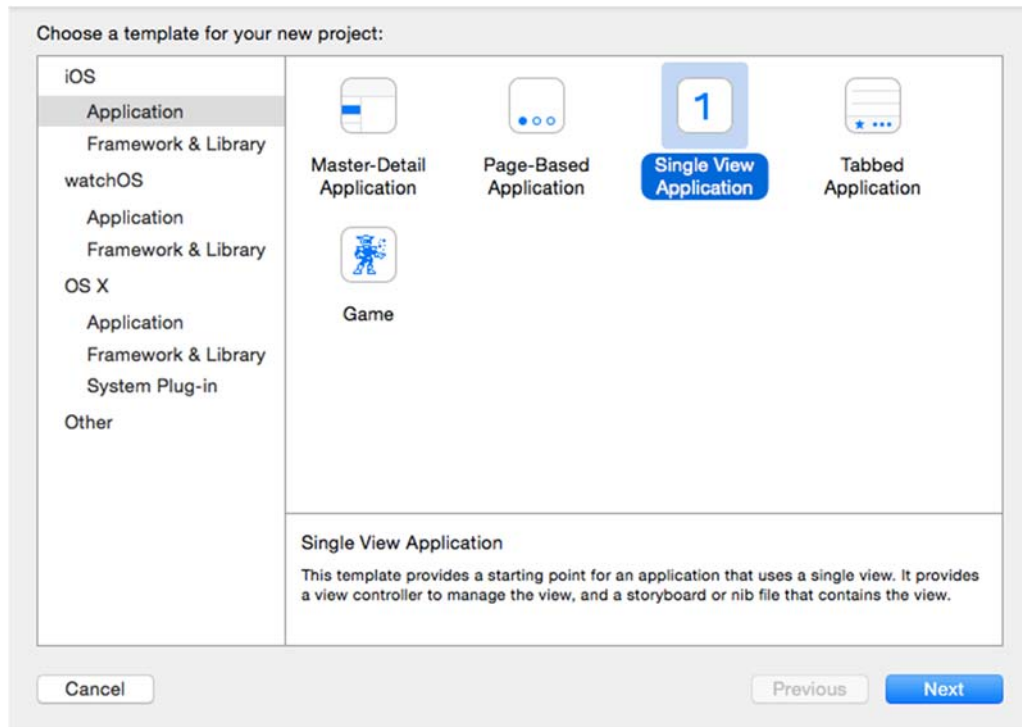


【執行結果】

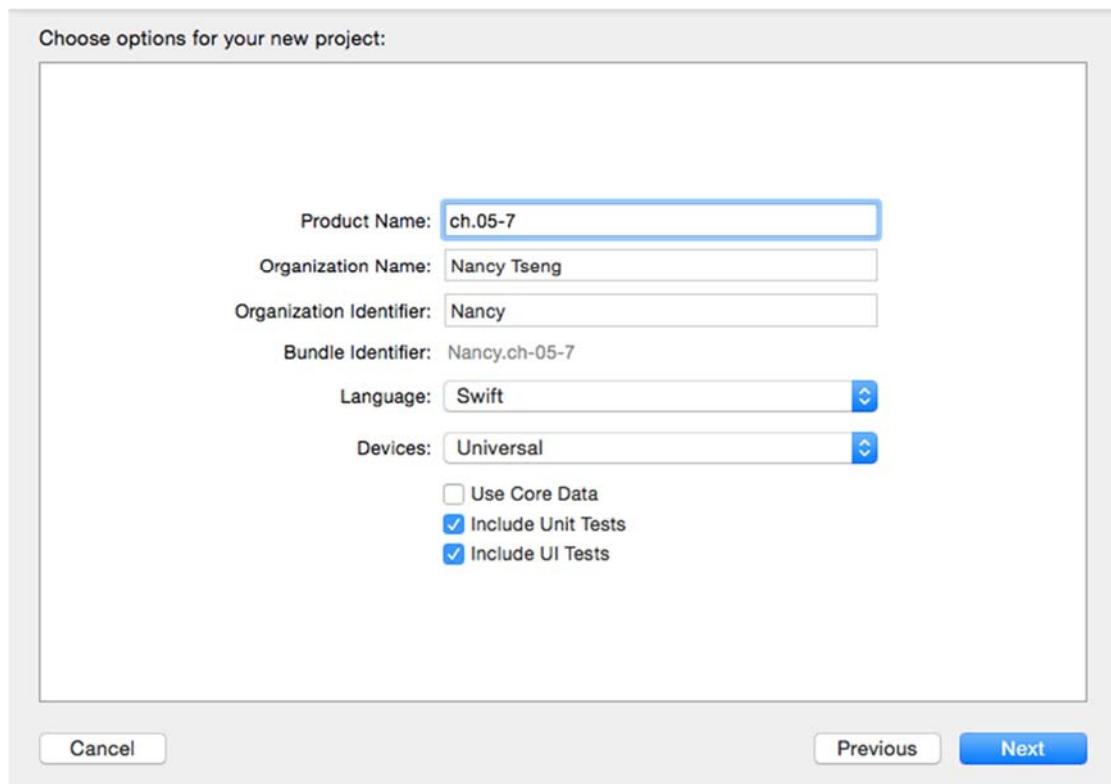
當 App 執行後，操作者在滑動日期時會將移動結果，顯示數值在【文字標籤】中。

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 `ch.05-7`

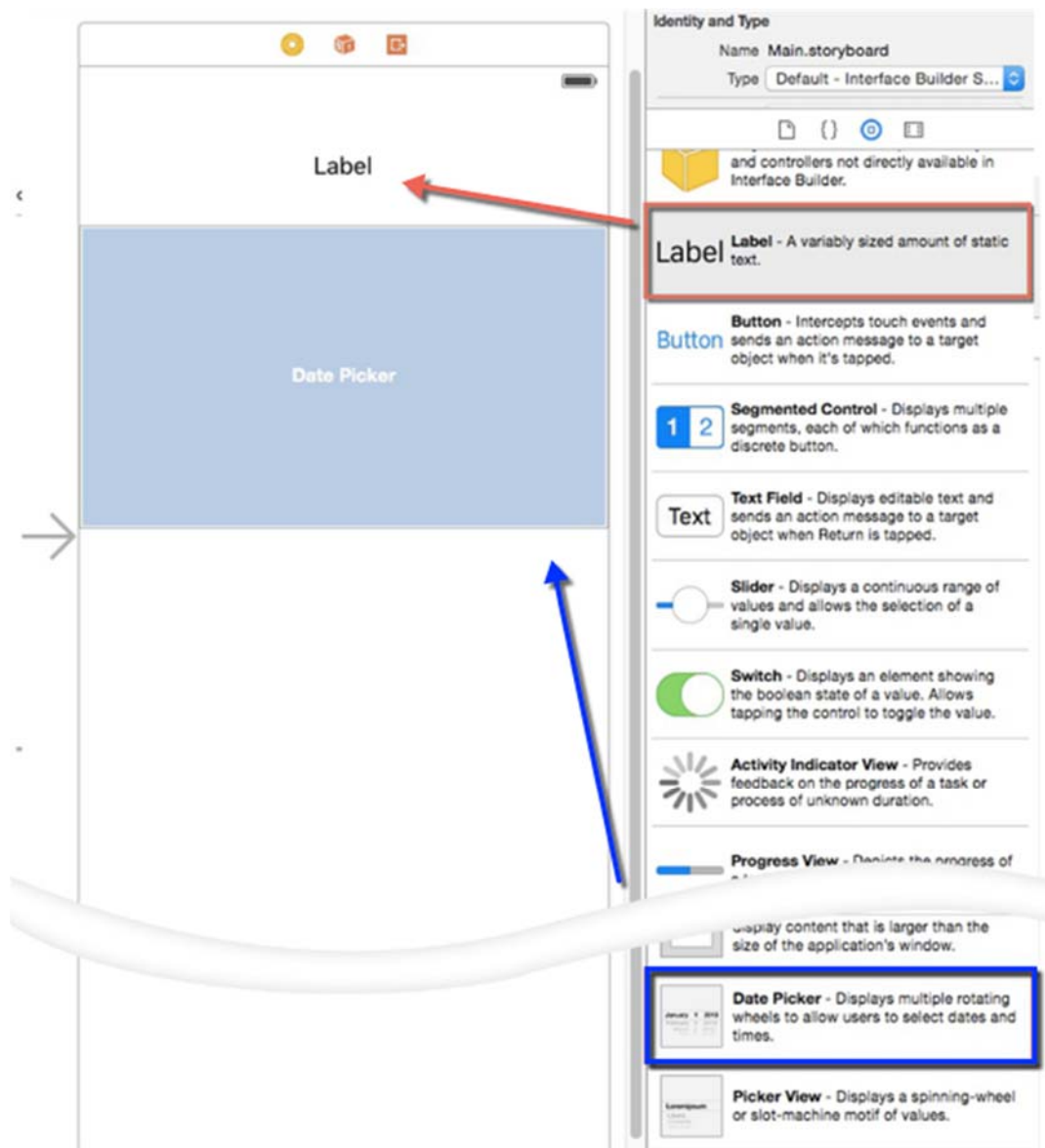


Step.2

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】及【Date Picker】到 IB 畫面中。



Step.3.

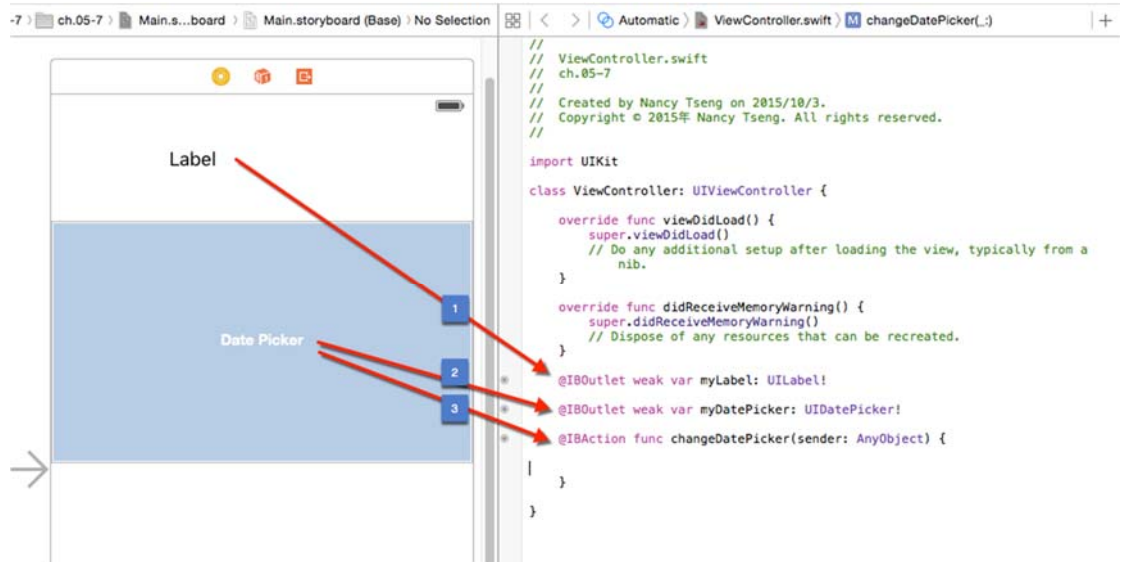
接著點選右上方工具列視窗【輔助編碼器】，就是雙圈符號【2】，進行程式碼編輯。



(詳見 5-1 屬性設定小技巧)

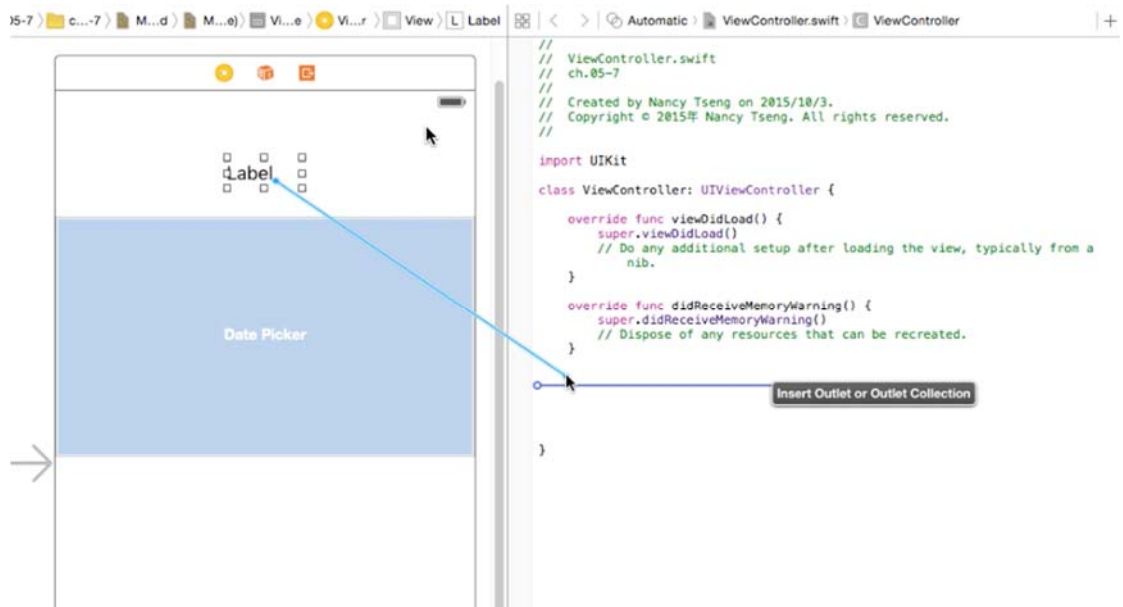
Step.4

- (1) 透過連結即產生【紅框】處的程式碼，控制 IB 建立的元件和【程式碼】連結。



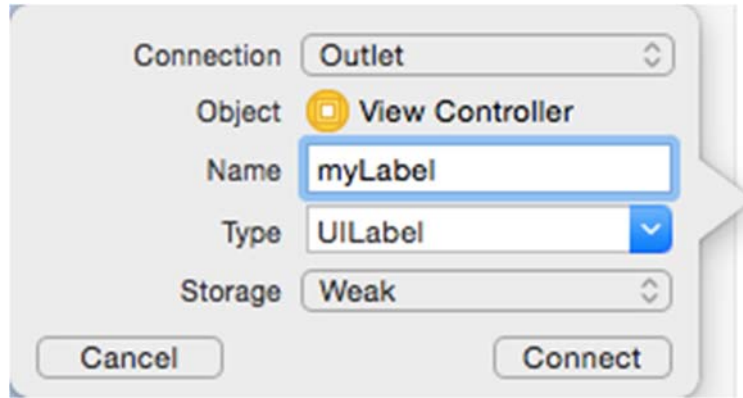
分解如下

- (2) 將【文字標籤】【方框一】與「變數名稱」連結。



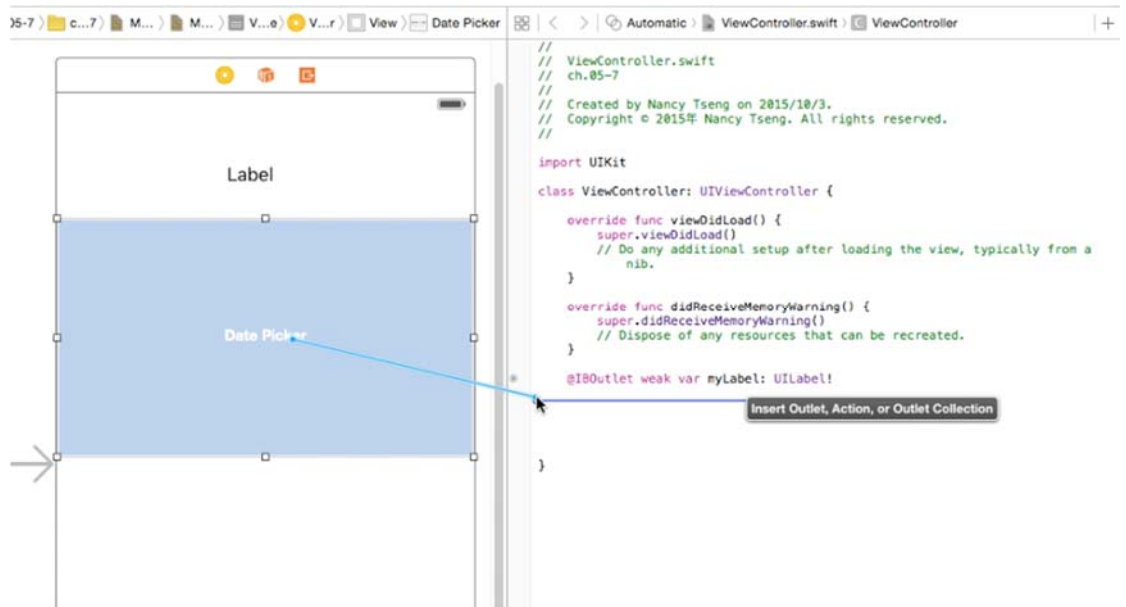
拖曳 Lable 元件

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



在拖曳後自動彈跳的視窗

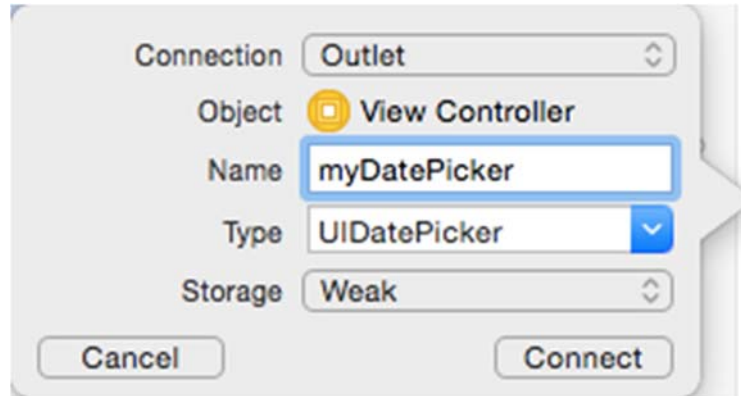
(3) 將【日期時間方塊】【方框二】及【方框三】與「變數名稱」連結。



拖曳 Date Picker 元件，分別各設【Outlet】、【Action】

【方框二】

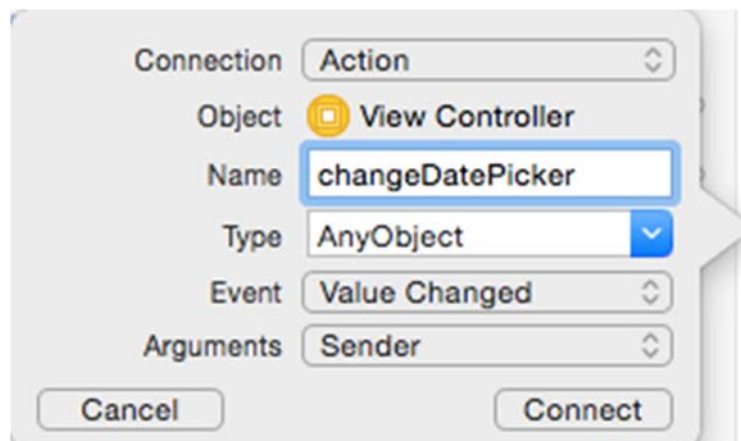
在「Connection」欄位點選【Outlet】、「Name」欄位設定名稱【myDatePicker】以及在「Type」欄位點選【UIDatePicker】後，按【Connect】按鈕。



在拖曳後自動彈跳的視窗

【方框三】

在「Connection」欄位點選【Action】、「Name」欄位設定名稱【changeDatePicker】以及在「Type」欄位點選【AnyObject】後，按【Connect】按鈕。



(4) 在【方框一】、【方框二】及【方框三】，將會自動手插程式碼，作為與 IB 的連結。

```
@IBOutlet weak var myLabel: UILabel!  
@IBOutlet weak var myDatePicker: UIDatePicker!  
@IBAction func changeDatePicker(sender: AnyObject) {  
  
}
```

連結後出現的程式碼。

Step.5

- (1) 最後，我們在【紅框】中加入設定 **swift** 的程式碼，將透過串接後顯示在【文字標籤】裡。

```
import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view, typically from a nib.
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }

    @IBOutlet weak var myLabel: UILabel!

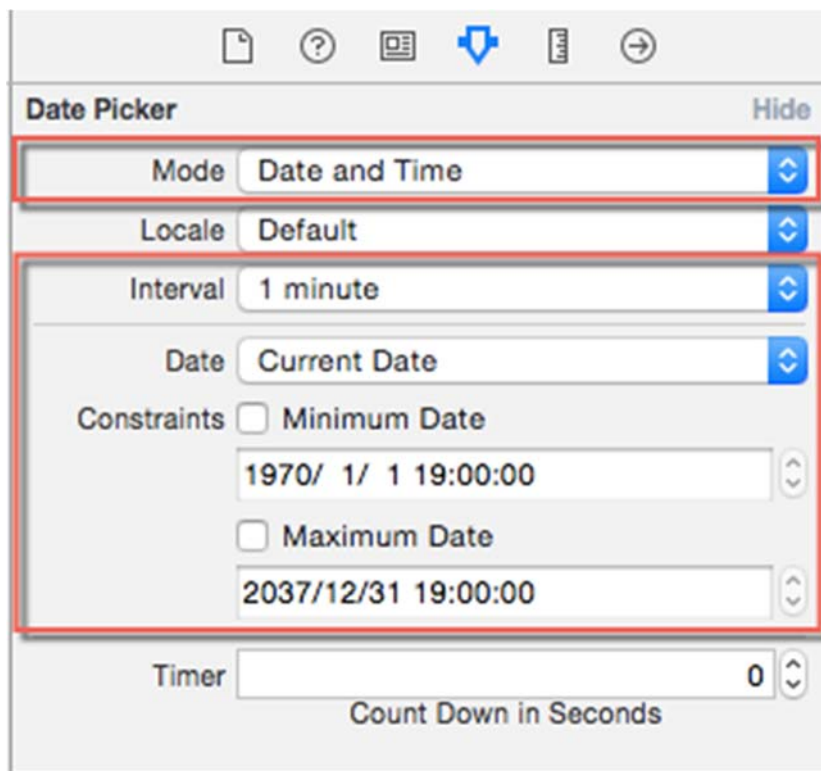
    @IBOutlet weak var myDatePicker: UIDatePicker!

    @IBAction func changeDatePicker(sender: AnyObject) {
        let df = NSDateFormatter()
        df.dateFormat = "yyyy/MM/dd HH:mm:ss"
        myLabel.text = df.stringFromDate(myDatePicker.date)
    }
}
```




屬性設定小技巧

關於【Date Picker】常用屬性說明：



選 項		功 能 說 明
1.	Mode	設定日期的型式
2.	Interval	設定時間間隔
3.	Date	設定預設日期選取
4.	Minimum Date	設定顯示起始日期值
5.	Maximum Date	設定顯示終端日期值

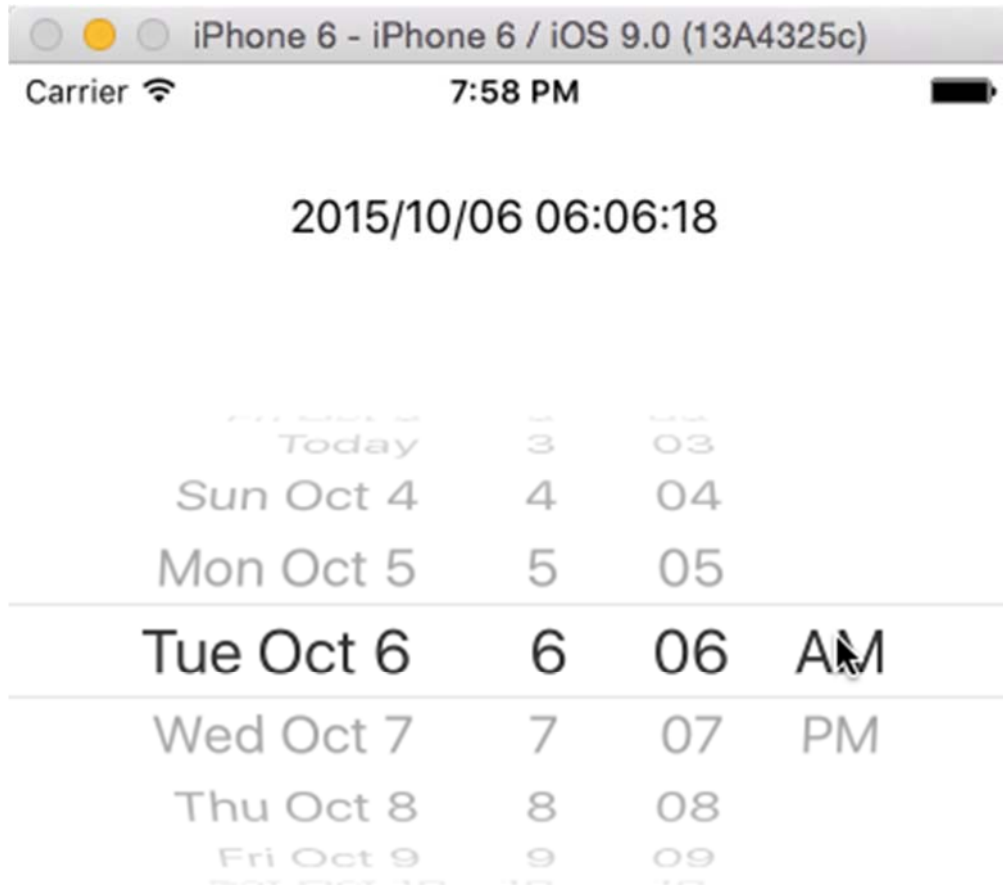
Step.6

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，在滑動【Date Picker】時間，將會自動帶出所設定的內容，顯示在【文字標籤】也就是「Label」中的數值。

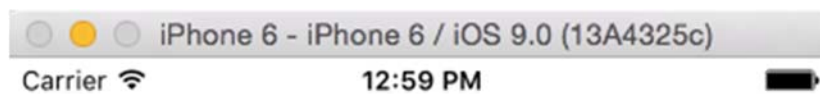


（小編在這裡設定為 iPhone6 畫面，供各位讀者們參考）

自我練習

實作執行後結果：

利用【Label】、【Button】及【Date Picker】程式結合執行，透過【Button】指令選項讓【Label】依據【Date Picker】顯示內容產生我們想要的變化；分別顯示「日期」、「時間」、「日期及時間」以及「目前的時間」這四者在與【程式碼】連結並串接時，當執行【RUN】後所顯示數值內容。



2015/10/04 19:59

Date	Time	DateTime	
Fri Oct 2	5	57	
Sat Oct 3	6	58	AM
Today	7	59	PM
Mon Oct 5	8	00	
Tue Oct 6	9	01	

Now