

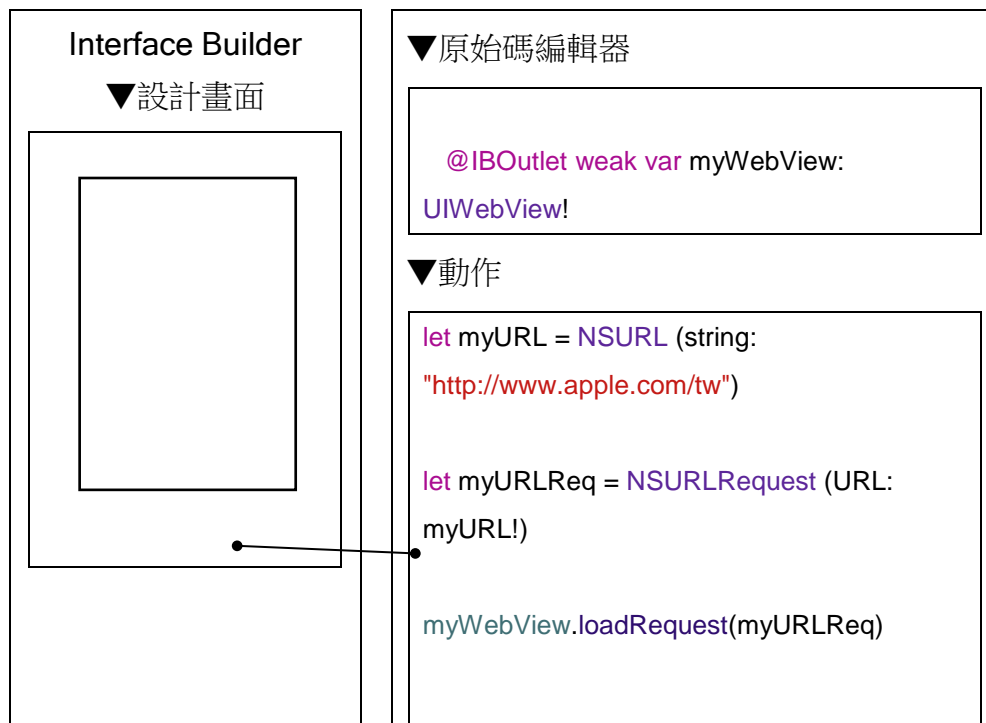
## CHAPTER 8-1

### UIWebView: 顯示網頁

#### 將網頁內容顯示於網頁視窗

學習概念：

1. 首先用 IB 建立【UIWebView】。
2. 接著將【UIWebView】與【程式碼連結】。
3. 最後在實作相關程式，載入網頁內容

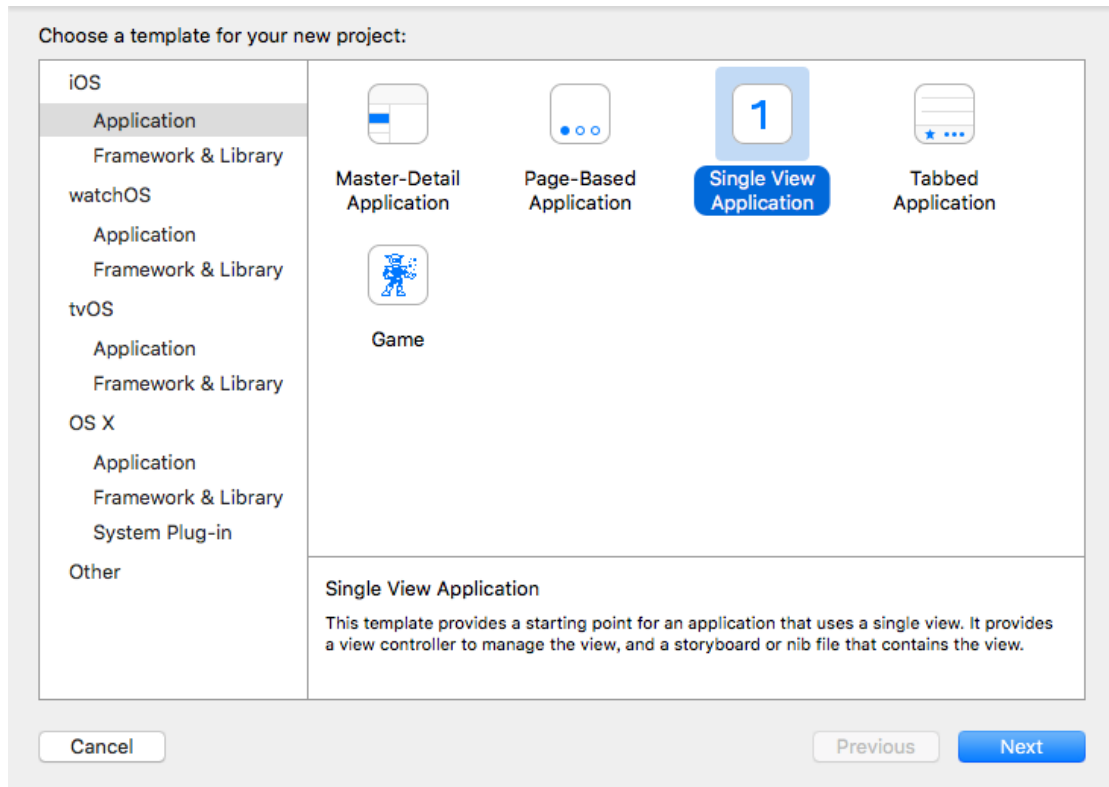


#### 【執行結果】

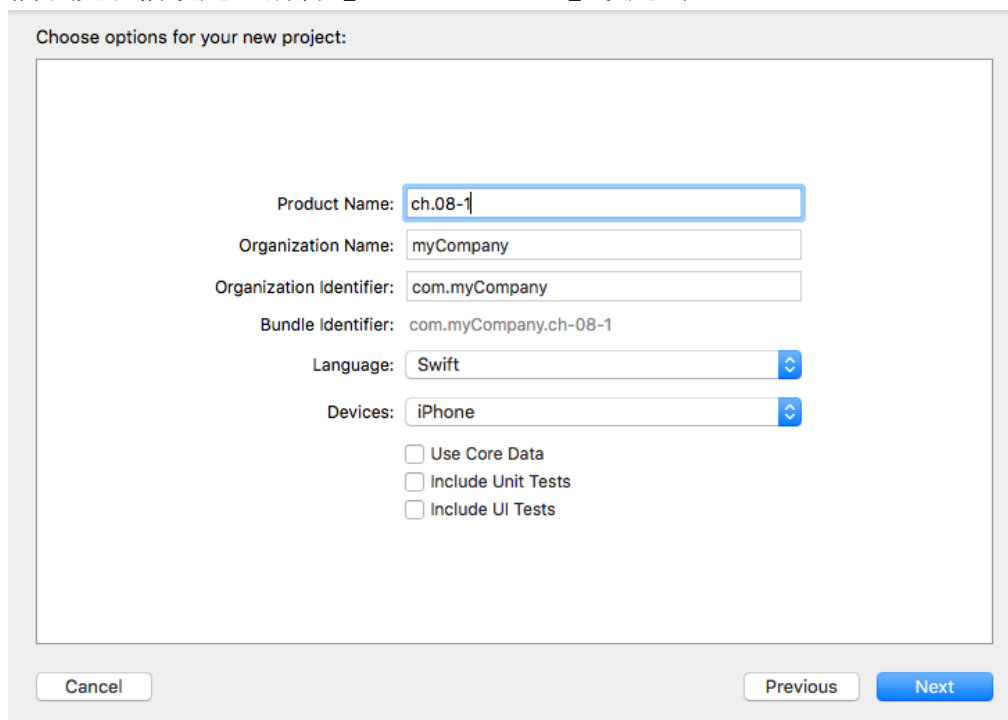
當 App 執行後，會自動於【網頁視窗】上開啟設定的網頁。

## 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。

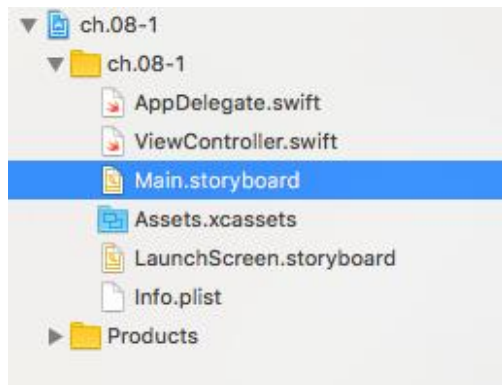


檔名及名稱設定，請將【**Product Name**】設定為 **ch.08-1**

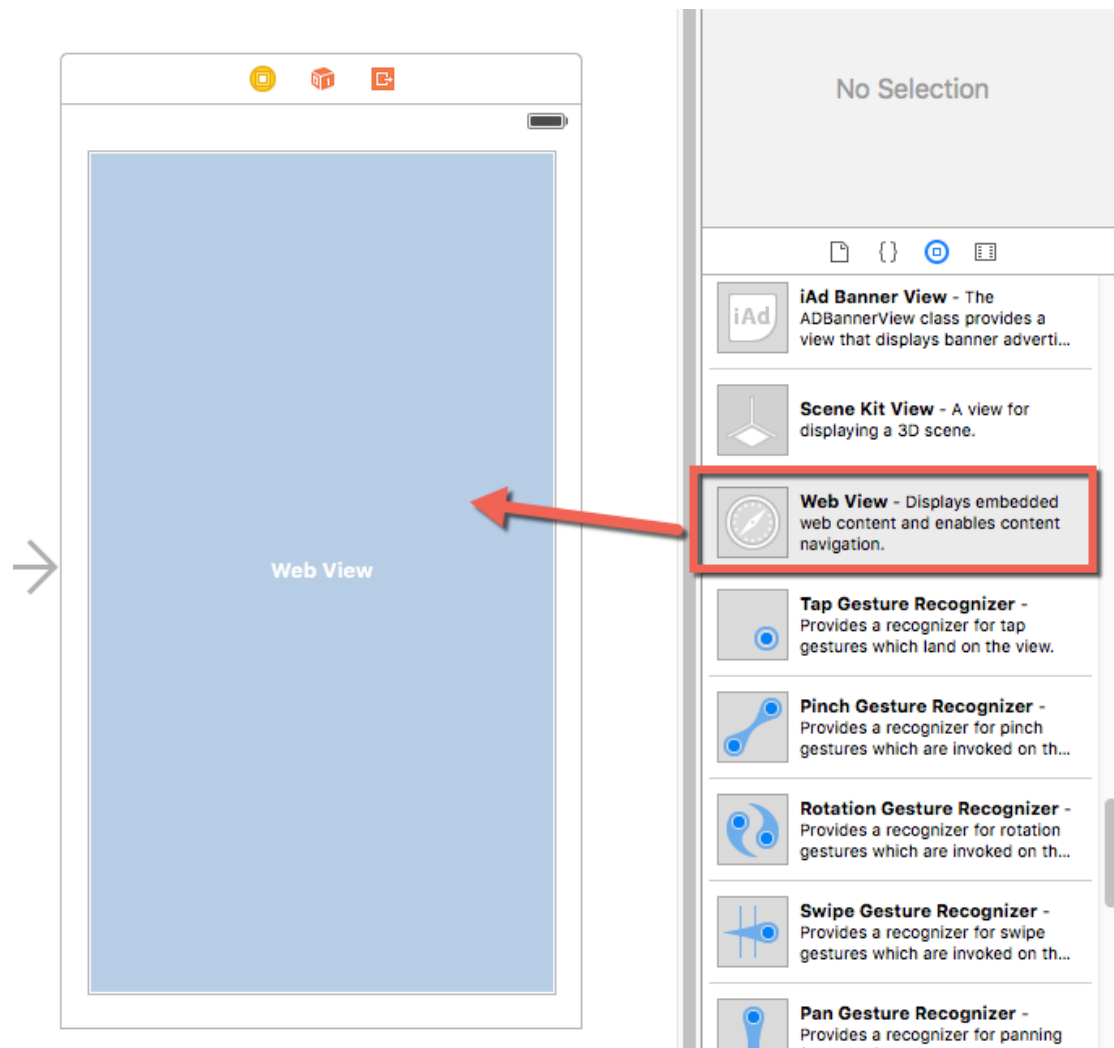


## Step.2

選取【Main.storyboard】

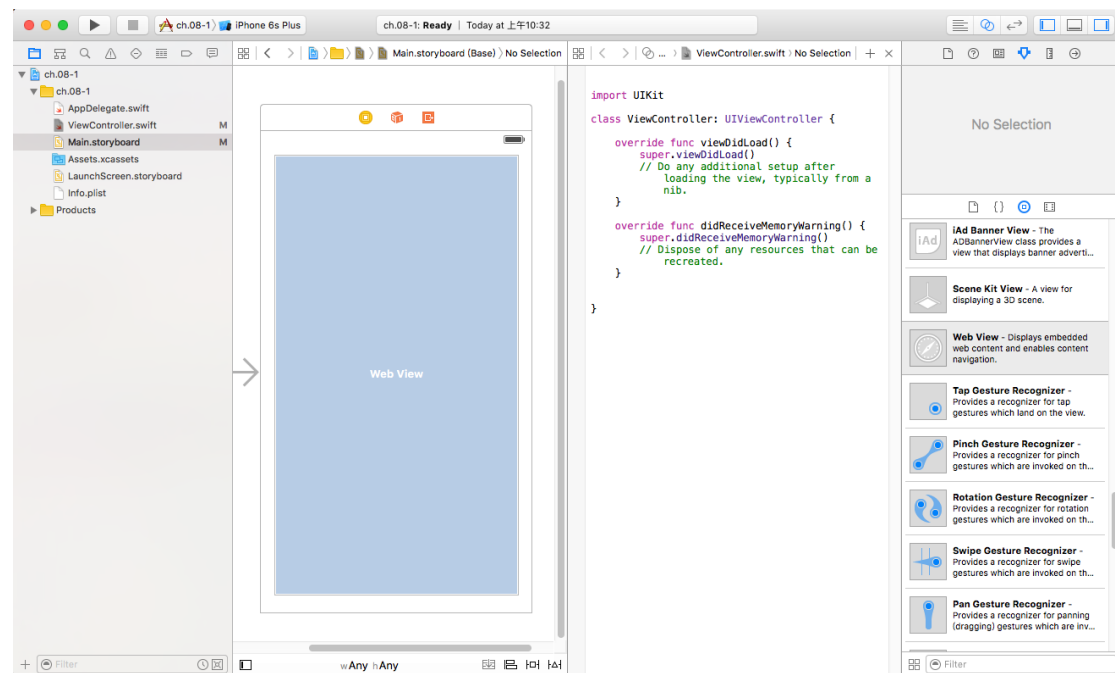


從【物件庫】中拖曳【Web View】到畫面中。



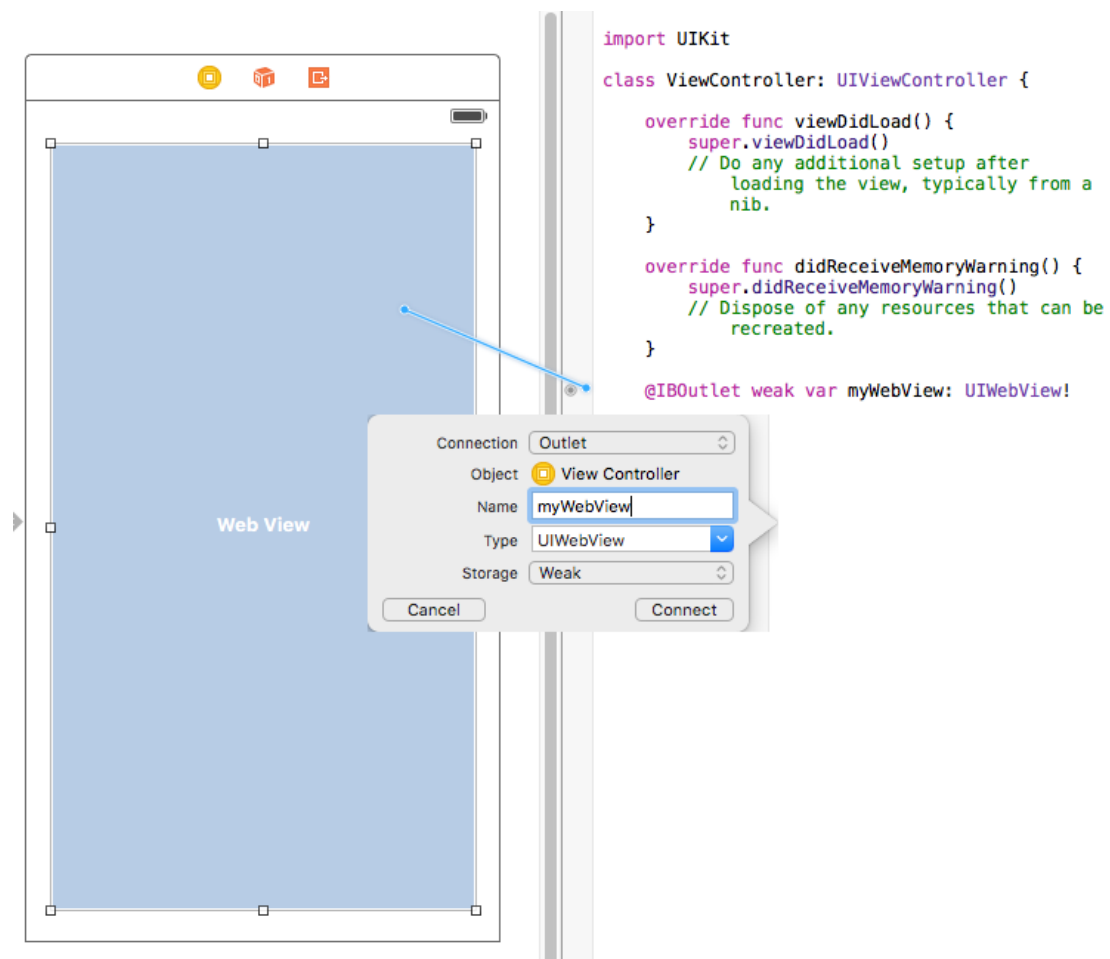
## Step.3.

將編輯畫面分割為四部分，以方便連結元件與程式



## Step.4

(1) 將【Web View】與程式連結，命名 myWebView



## Step.5

(1) 建立 URL 物件。

```
let myURL = NSURL (string: "http://www.apple.com/tw")
```

(2) 建立 URLRequest 請求物件。

```
let myURLRequest = NSURLRequest (URL: myURL!)
```

(3) 將 URLRequest 請求物件建立 URL 物件。

```
myWebView.loadRequest(myURLRequest)
```

```
import UIKit

class ViewController: UIViewController {

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

        let myURL = NSURL (string: "http://www.apple.com/tw")

        let myURLRequest = NSURLRequest (URL: myURL!)

        myWebView.loadRequest(myURLRequest)

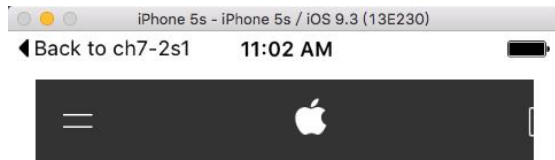
    }

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

    @IBOutlet weak var myWebView: UIWebView!
}
```

## Step.6

執行程式。



# Apple WATCH

一目了然，就是你。

[進一步了解](#) > [觀看 keynote 發表會](#) 



# 自我練習

實作執行後結果：

於畫面中，上方加入文字框，可供輸入網址，輸入網址後按下 Go 的按鈕，即可在下方的 **web view** 中載入網頁。

