

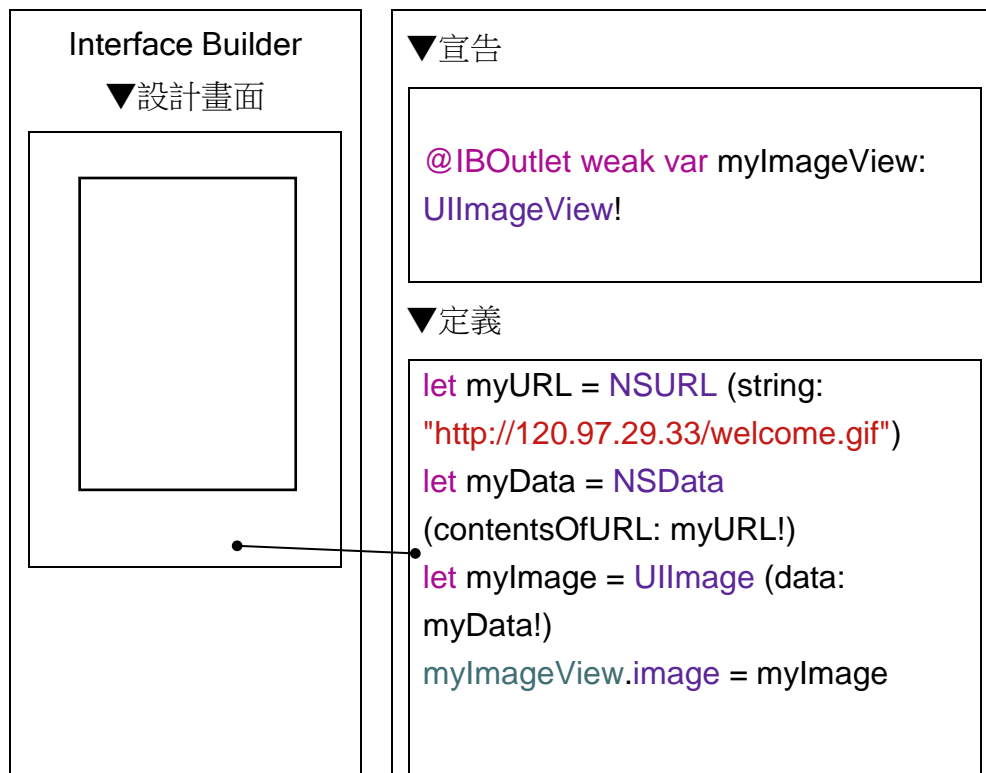
CHAPTER 8-2

UIWebView: 顯示網頁的圖片

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. 在處理載入網頁的事件。

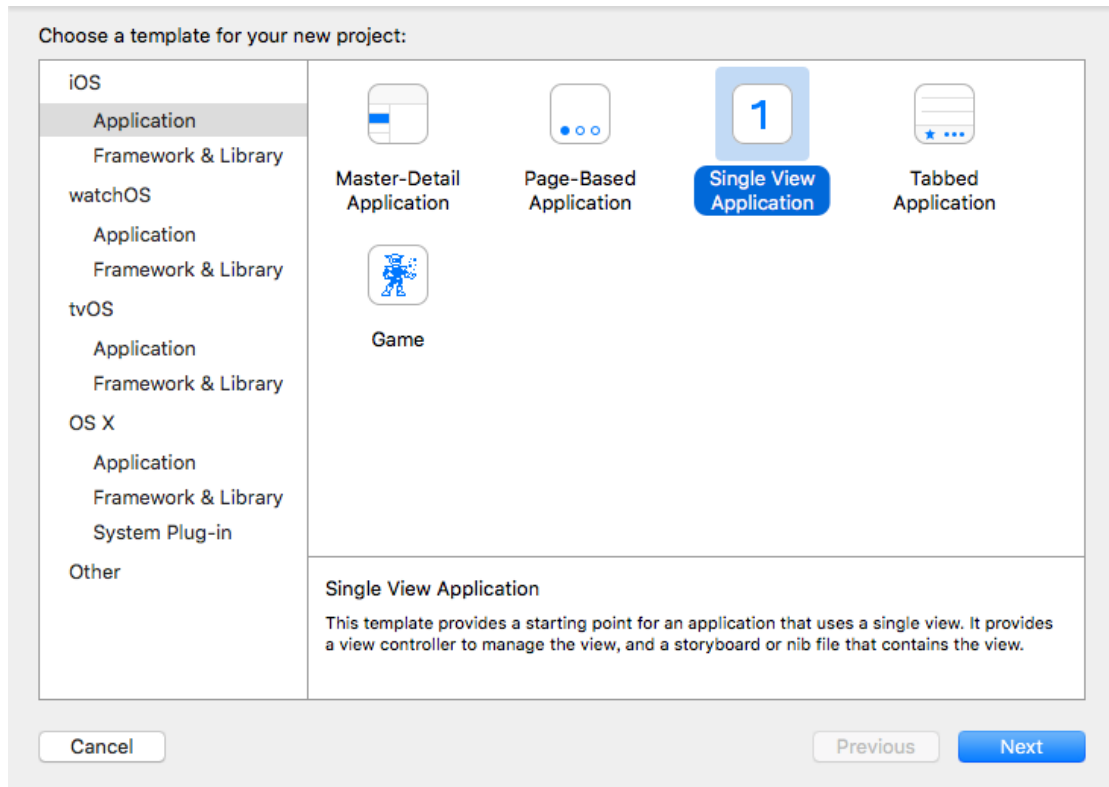


【執行結果】

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.08-2**

Choose options for your new project:

Product Name:

Organization Name:

Organization Identifier:

Bundle Identifier:

Language:

Devices:

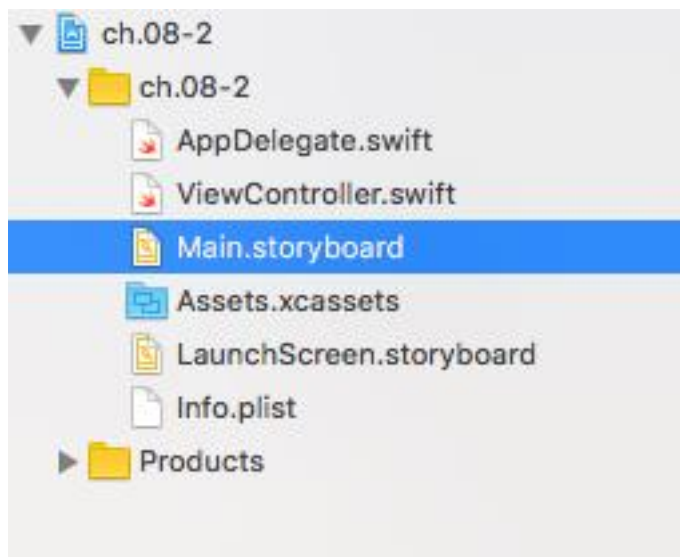
☐ Use Core Data

☐ Include Unit Tests

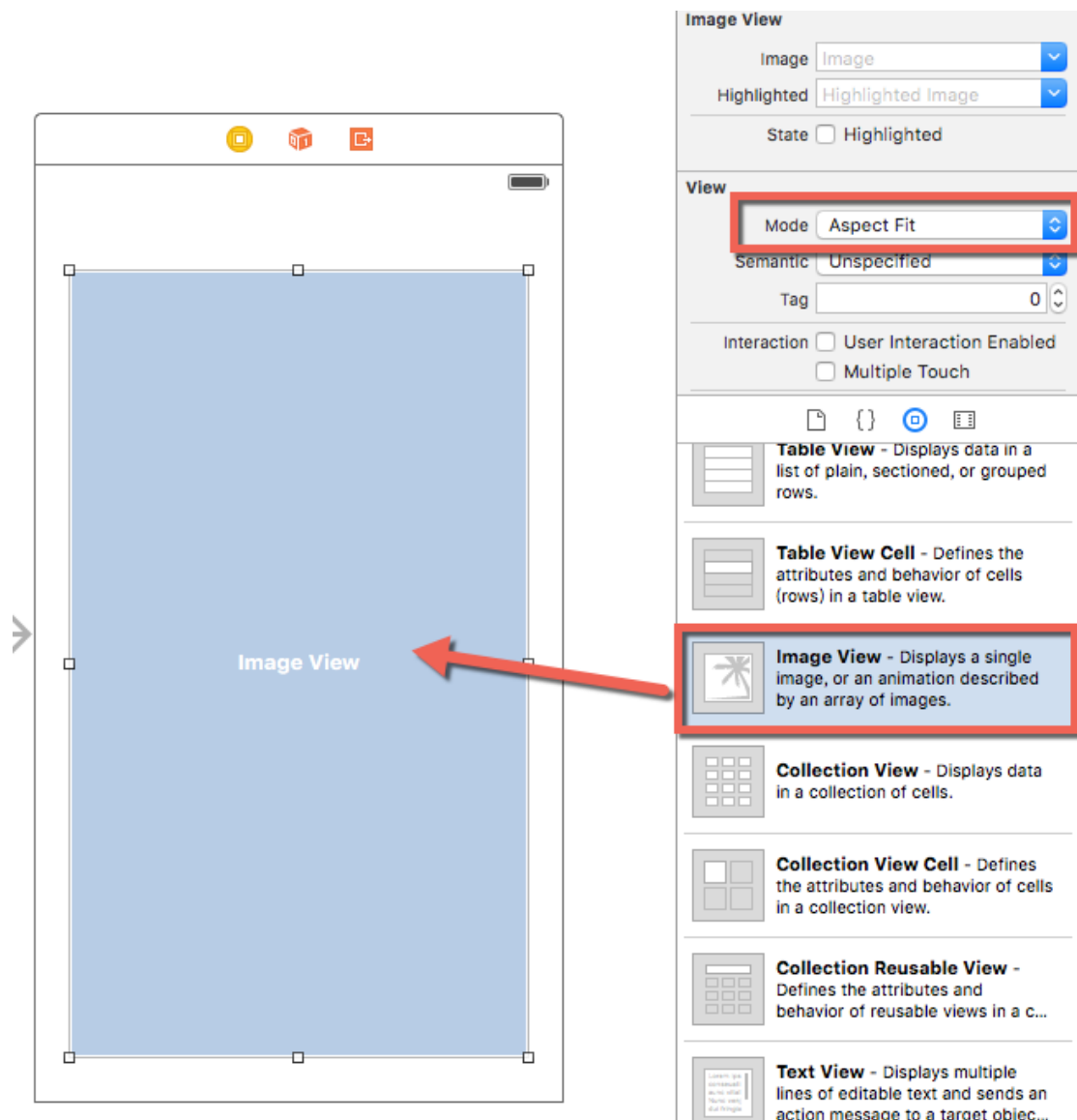
☐ Include UI Tests

Step.2

選取【Main.storyboard】

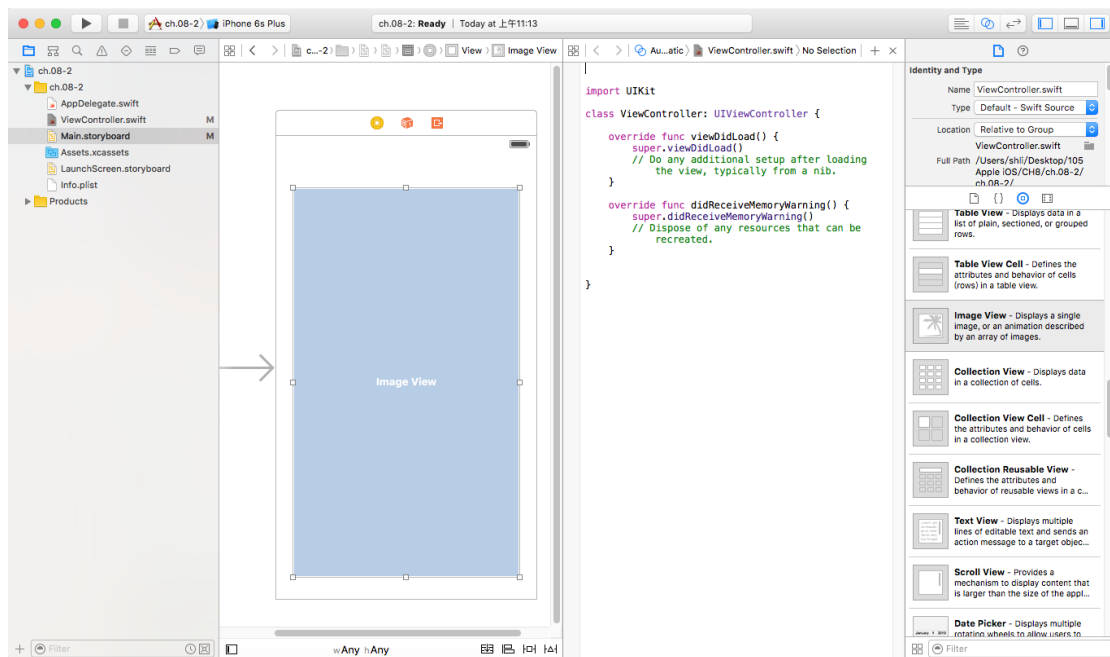


從【物件庫】中拖曳【Image View】到畫面中，並設定 mode 為 Aspect Fit，避免載入的圖片變形。



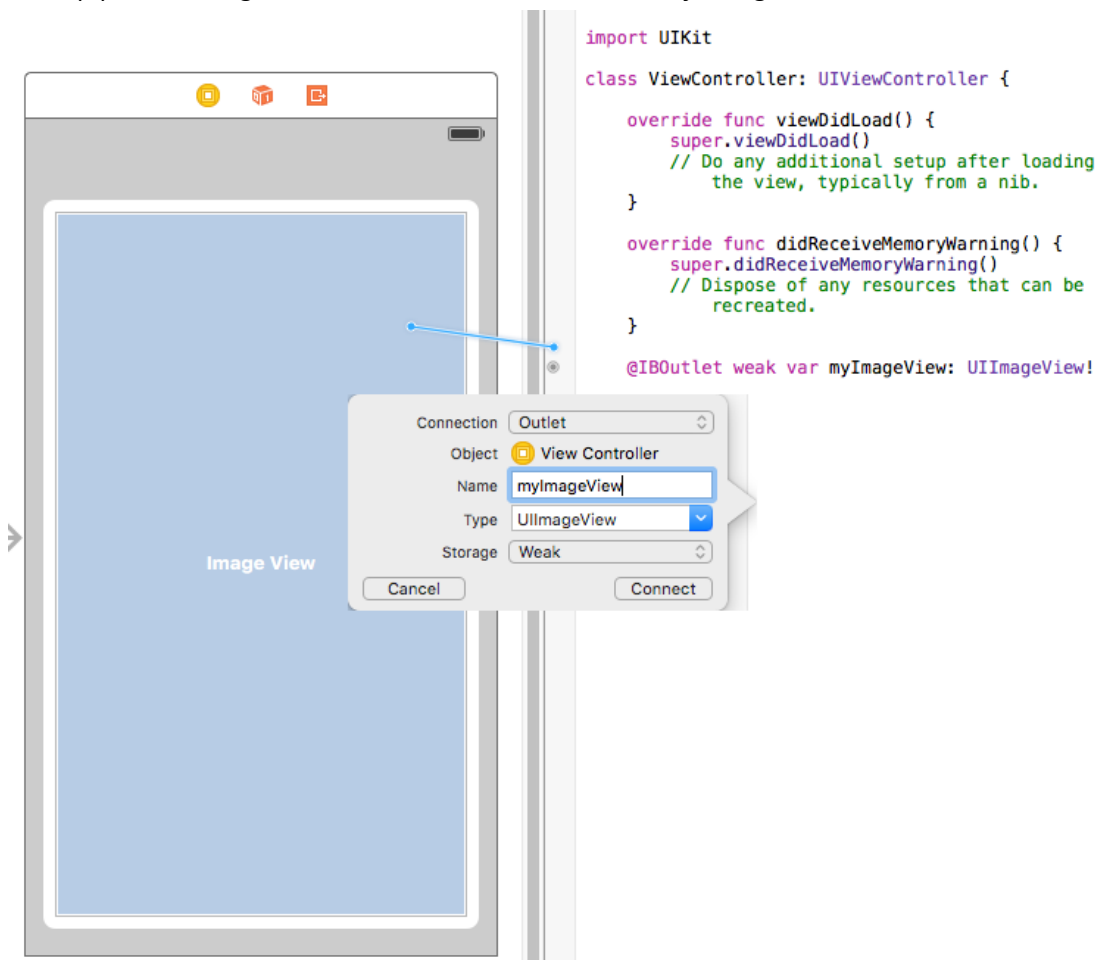
Step.3.

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【Image View】與程式連結，命名 myImageView



Step.5

(1) 建立 URL 物件。

```
let myURL = NSURL (string: "http://120.97.29.33/welcome.gif")
```

(2) 由 myURL 網址中取出資料，並將此料儲存於 myData 中。

```
let myData = NSData (contentsOfURL: myURL!)
```

(3) 將取出的資料以圖像的資料型別儲存。

```
let myImage = UIImage (data: myData!)
```

(4) 將 URLRequest 請求物件建立 URL 物件。

```
myImageView.image = myImage
```

```
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://120.97.29.33/welcome.gif")
        let myData = NSData (contentsOfURL: myURL!)
        let myImage = UIImage (data: myData!)
        myImageView.image = myImage

    }

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

    @IBOutlet weak var myImageView: UIImageView!
}
```

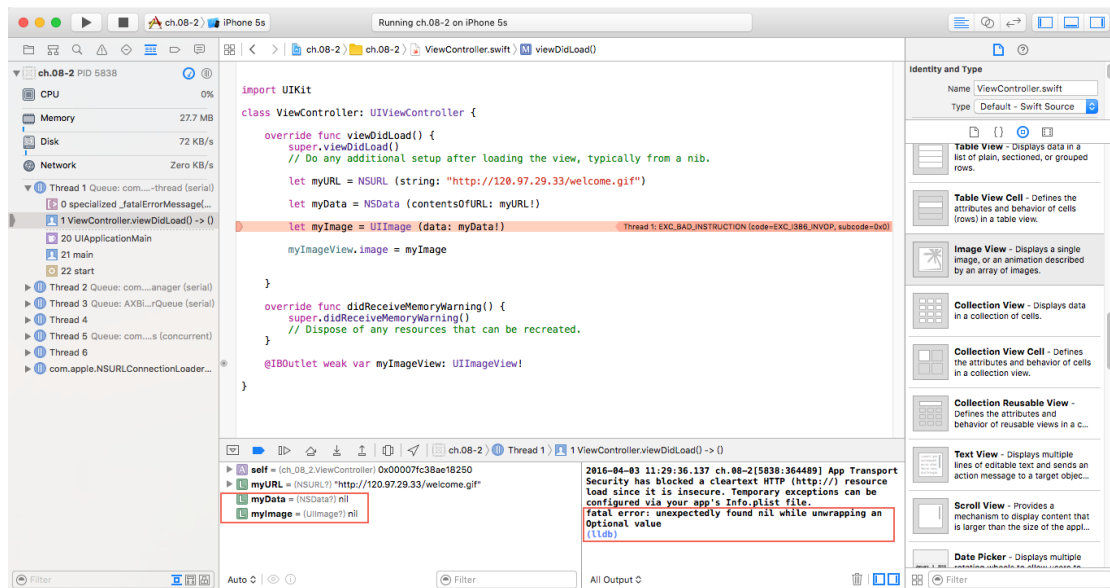
Step.6

執行程式。

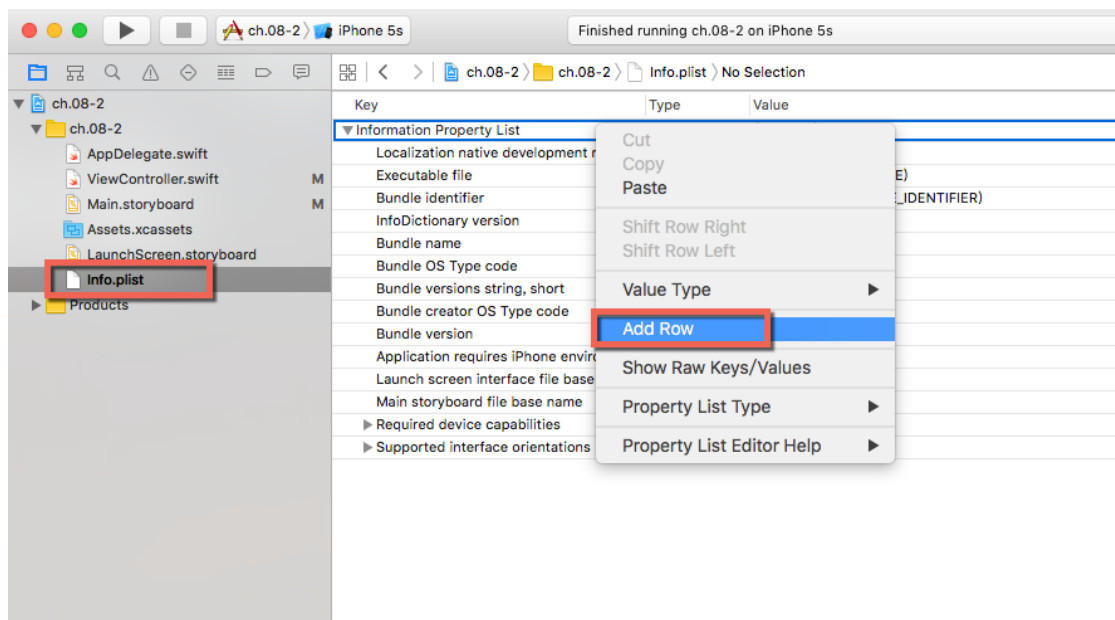
此時會跳出錯誤訊息

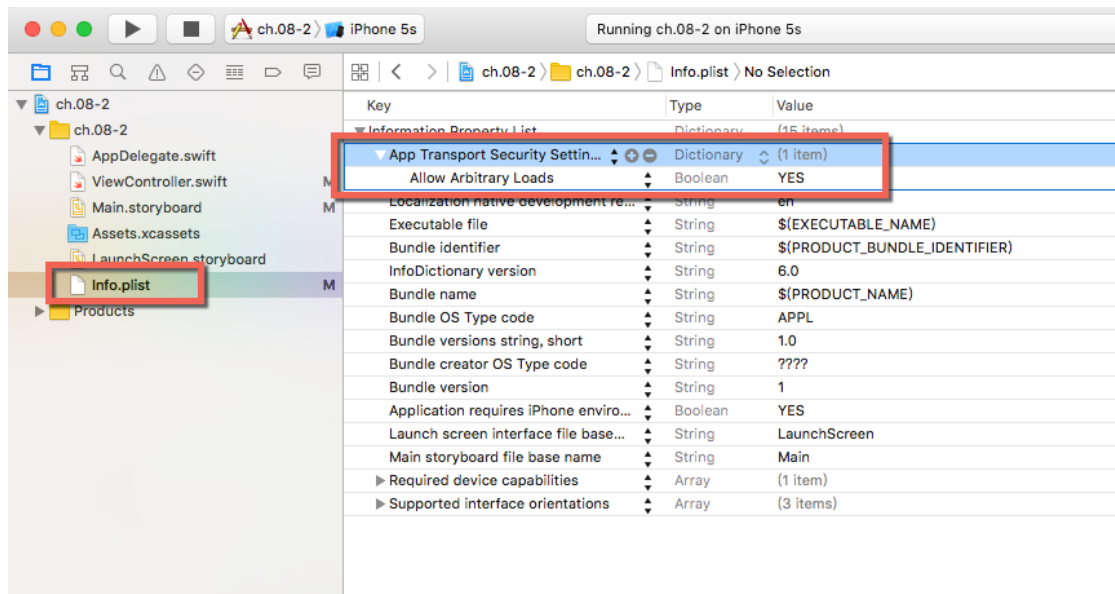
由左下方的變數現值中可以看到， myURL 有值，可是到 myData 時卻無法取得資料，但是此網址確實是有該圖片。

原因出在 apple 在 iOS 9 之後預設只能開啟 https 的網頁取值



在 Info.plist 中添加 App Transport Security Settings，並於項下添加 Allow Arbitrary Loads 類型為 Boolean,將值改為 YES





Step.7

重新執行程式，即可正常顯示圖片。



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 Enter 鍵，即可在下方的 Image view 中載入圖片。

技巧提示：必須使用到 TextField 的函式 `textFieldShouldReturn`，因此必須於 `ViewController` 中加入 `UITextFieldDelegate`。並且於程式執行時告知 `myTextField` 的作用要應用在此程式上。

```
class ViewController: UIViewController, UITextFieldDelegate {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        myTextField.delegate = self  
    }  
  
    func textFieldShouldReturn(textField: UITextField) -> Bool {  
        <#code#>  
    }  
}
```

<http://120.97.29.33/welcome.gif>

