

# iOS分享會(二)講義

## <一>語法復習與觀念

### Lab1:

- #1.產出一個NSString，內容為@"123.456.789.10"
- #2.練習查詢Method，將之Split為NSMutableArray
- #3.練習查詢Method，將Split出的NSMutableArray改變為[999.999.999.999]
- #4.把此NSMutableArray重組成NSString並Log結果

### Lab1練習目的：

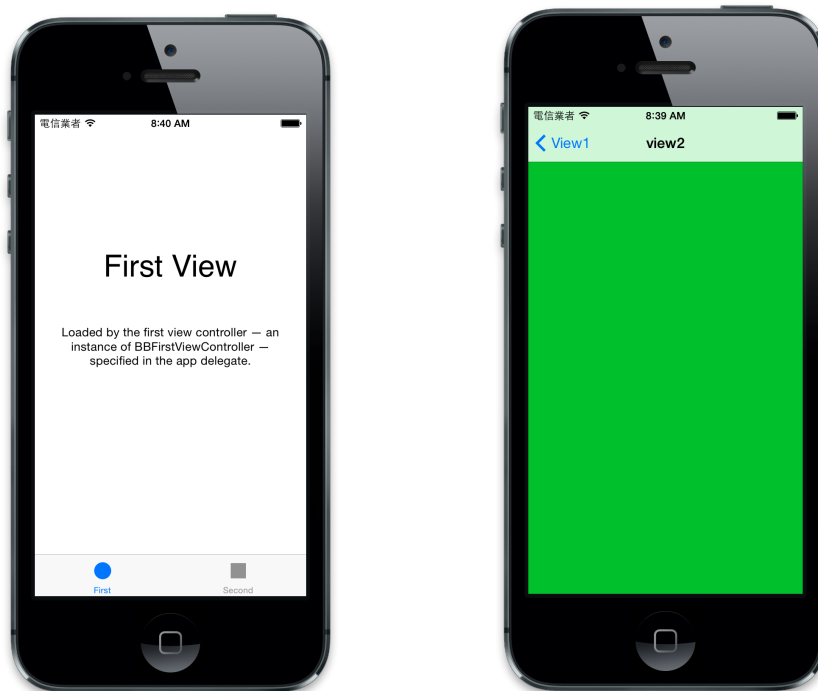
- 1.復習尋找自己需要的Method。
- 3.Obj-C中，一開始最難看懂多重中括號用法。  
(Demo :ZombieTest<SpriteKit>)
- 4.何謂深/淺層Copy，要深入了解的請爬書爬文。
- 5.了解Obj-C可與C語法混用。

## <二> 頁面架構

### 2-1. 何謂頁面架構？

頁面架構像是一個紙盒，可以想像他是一個裝A4紙盒，而可視頁面在其中則像是一張張A4紙。（我們由上方俯視）

### 2-2. 最常見的頁面架構 - UINavigationController、UITabBarController



### 2-3. 頁面管理記憶體架構。

### 2-4. 實做練習 - (TabTest & UINavigationControllerTest)。

- 2-4-1: 開啟一個TabBarController樣板。
- 2-4-2: 了解TabBarController使用方式。
- 2-4-3: 開啟一個單一頁面專案樣板。
- 2-4-4: 自行開啟專案練習使用TabBarController。
- 2-4-5: 練習NavigationController使用及建立方式。

### 2-5. StoryBoard 切換頁面相關名詞介紹



Manual Segue  
push  
modal  
custom  
Relationship Segue  
view controllers

### 2-6. 如果不用Storyboard拉，而是用Code寫呢

## <三>Delegate語法介紹

### 3-1:何謂Delegate?有何作用?

Delegate其實很多翻譯寫成『代理人機制』，這個說法其實會讓人很摸不著頭緒。。。其實就是類似Java Interface概念的想法，也就是讓外部物件(非限定Class)可以設定到需Callback物件內部，呼叫call back function的概念。通常一般我們用在 —

3-1:Callback機制。

3-2:撰寫模組時，讓外部可以自行控制相關設定的資料。(講解TableView時再提)。

3-2:實際裡的概念看看Java — DelegatePromote。

3-3:Obj-C - delegate Demo。

### Lab2:

#1.產出一個有兩個頁面的UINavigationController架構的App。

#2.使用Delegate Method監聽View2回到View1的時機。

#3.在同一Project實作Log一萬次『鳥哥很帥』(切忌更改Log內文)。

#4.試著在Log的時候換頁吧

## <四>Thread(執行序)概念介紹

4-1.什麼是Thread(執行序)?

4-2.請看Demo Project<DelegateDemo>中的Thread操作示範。

4-3.Thread最大重點？

4-3-1 :Any UI code must on Main thread!

4-3-2：增進UX體驗。

4-4.增進效能(Preload)。

4-5.What is thread safe?

## <五>實作系統Delegate Method

1.使用Delegate概念，實際操作一下最常用的UI(UITableView)吧。

2.Demo project。