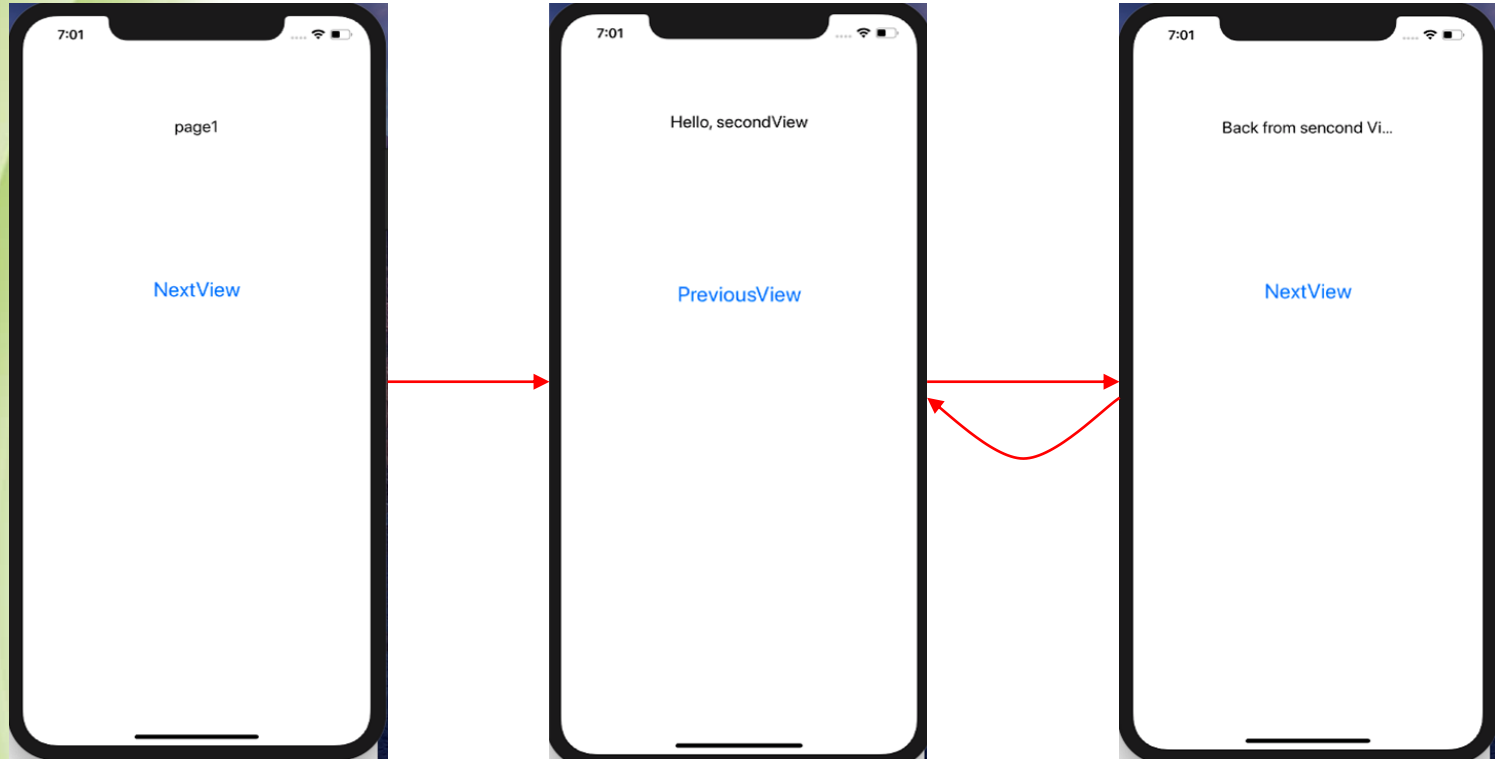


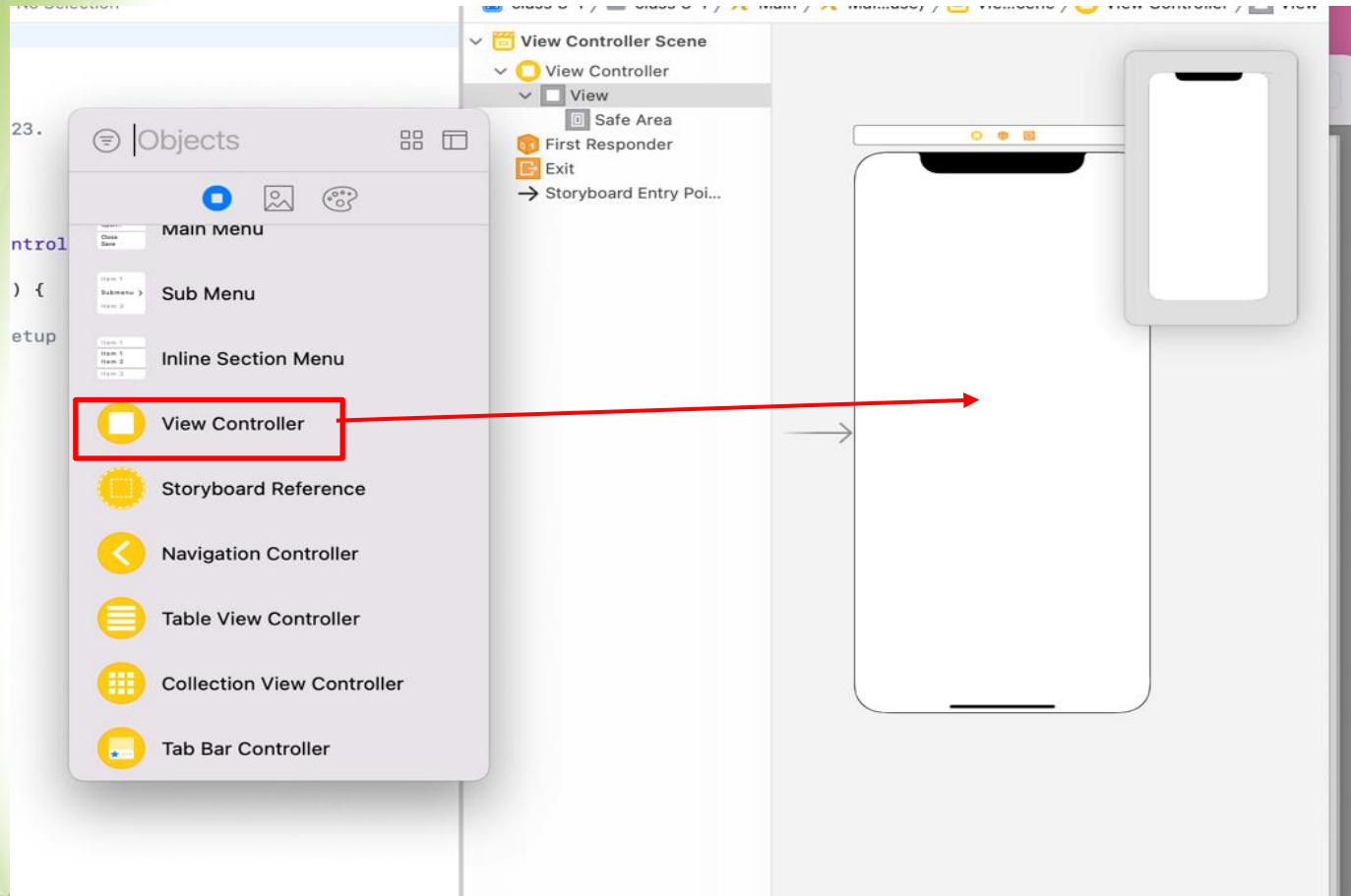
# Class 3



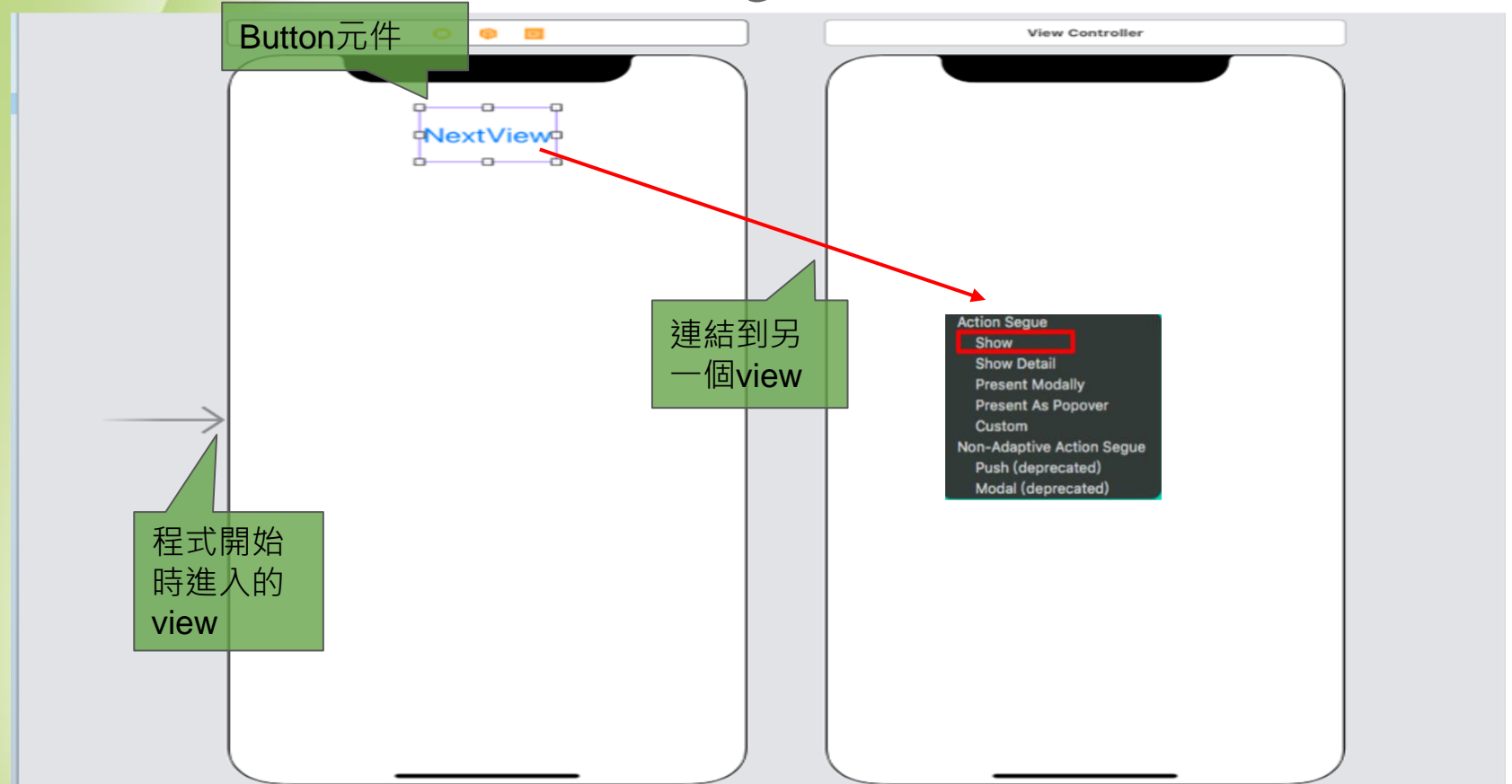
# Segue



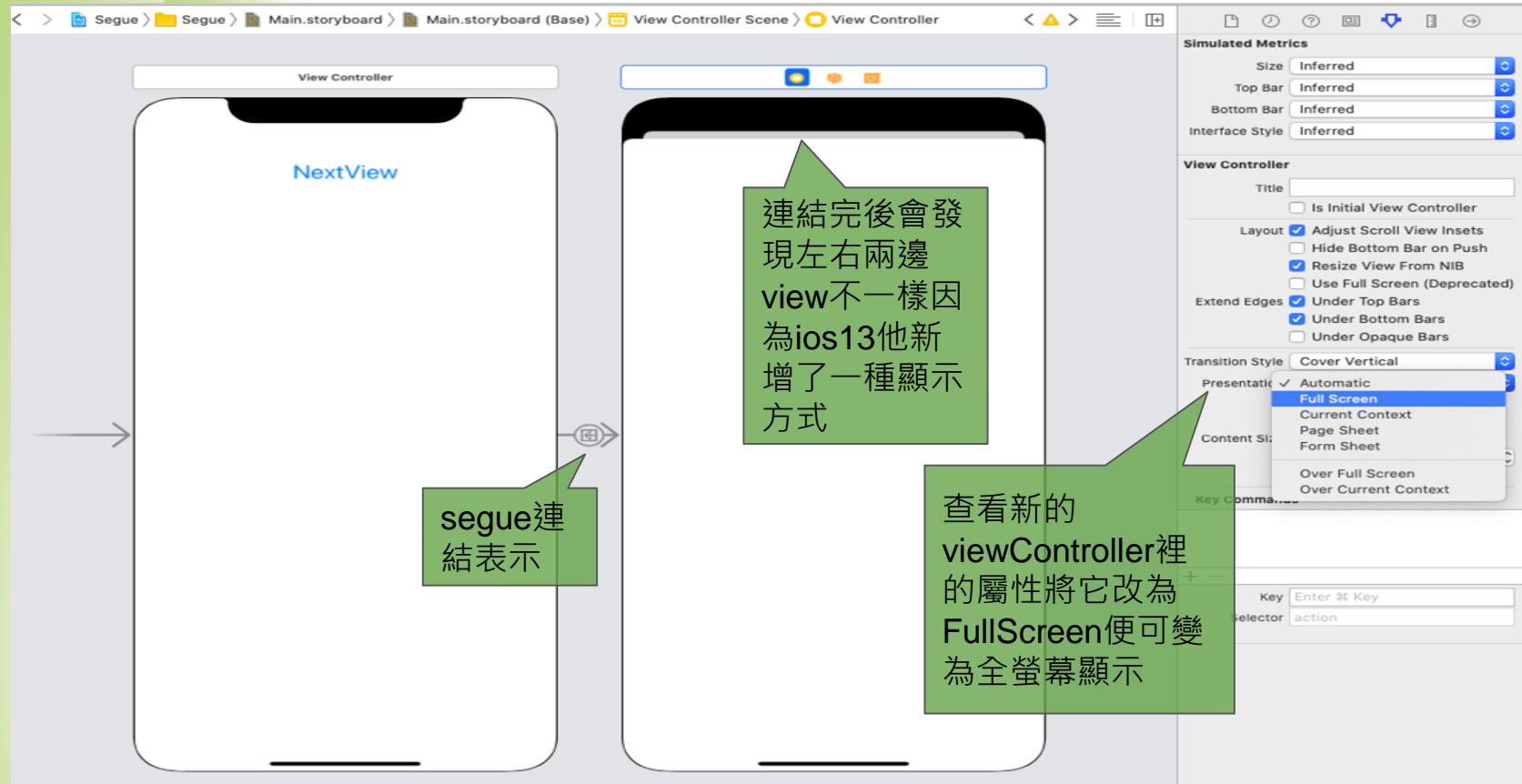
# Segue



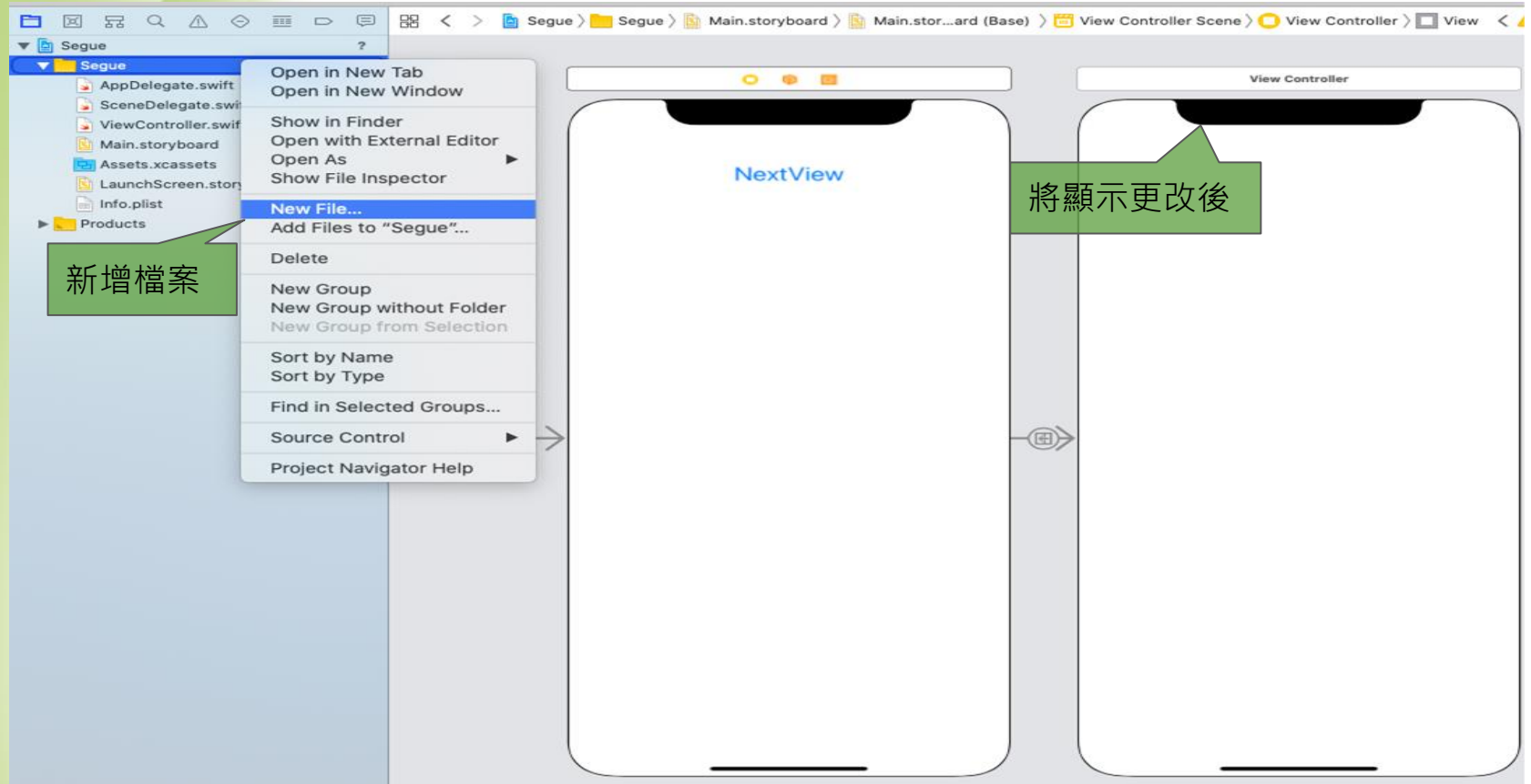
# Segue



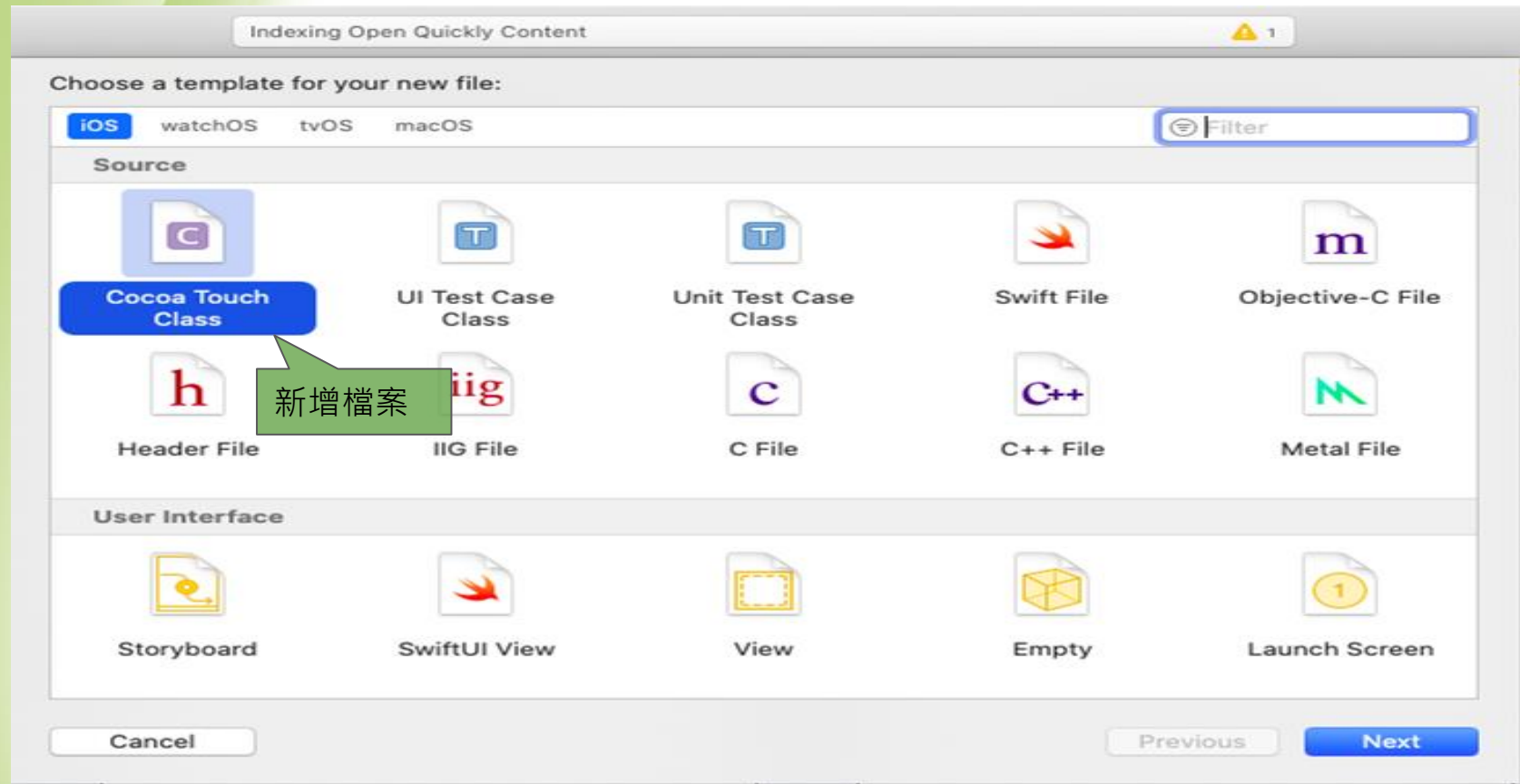
# Segue



# Segue



# Segue



# Segue

Choose options for your new file:

將檔案命名為第二  
個viewController

Class: secondViewController

Subclass of: UIViewController

☐ Also create XIB file

Language: Swift

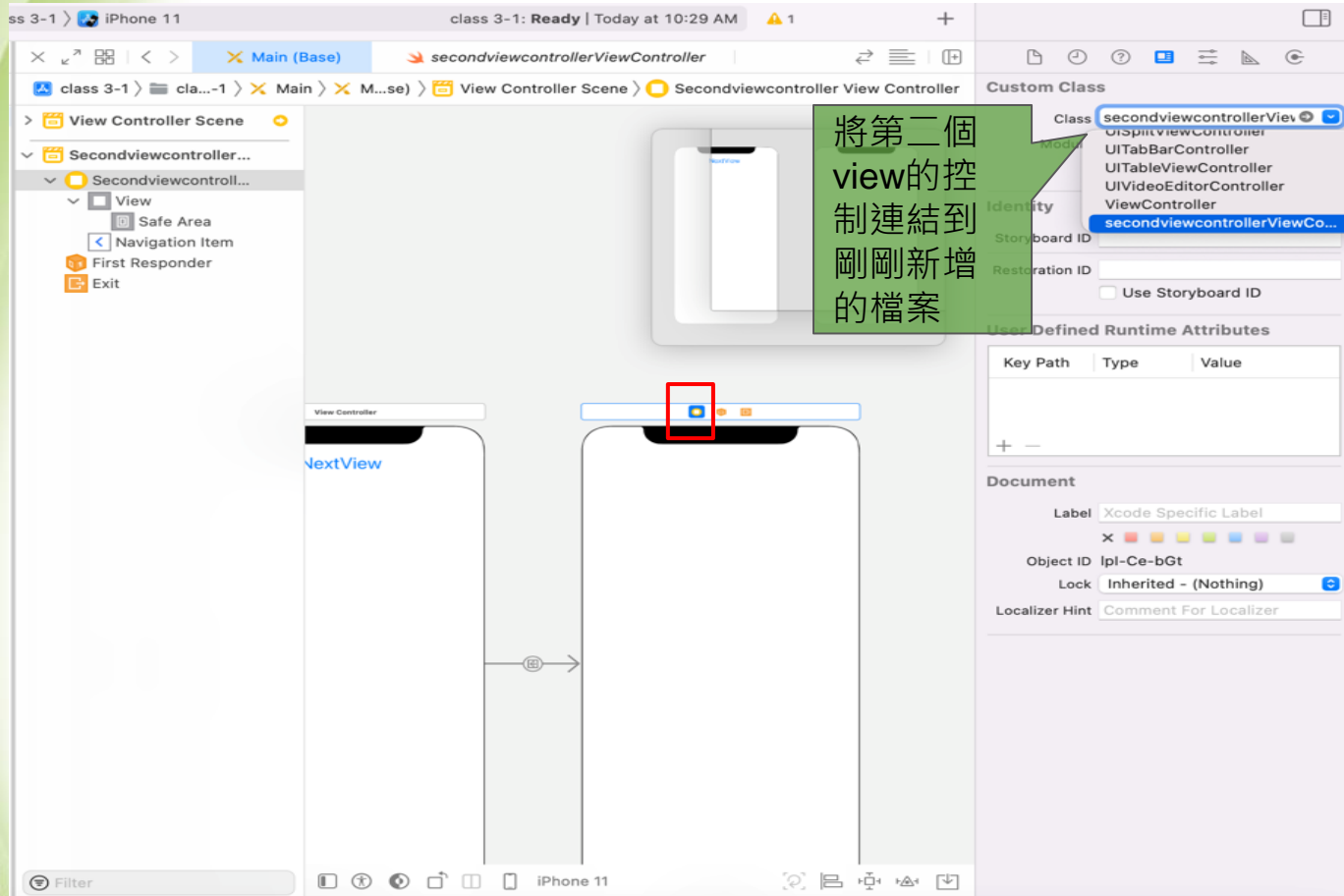
Cancel

Previous

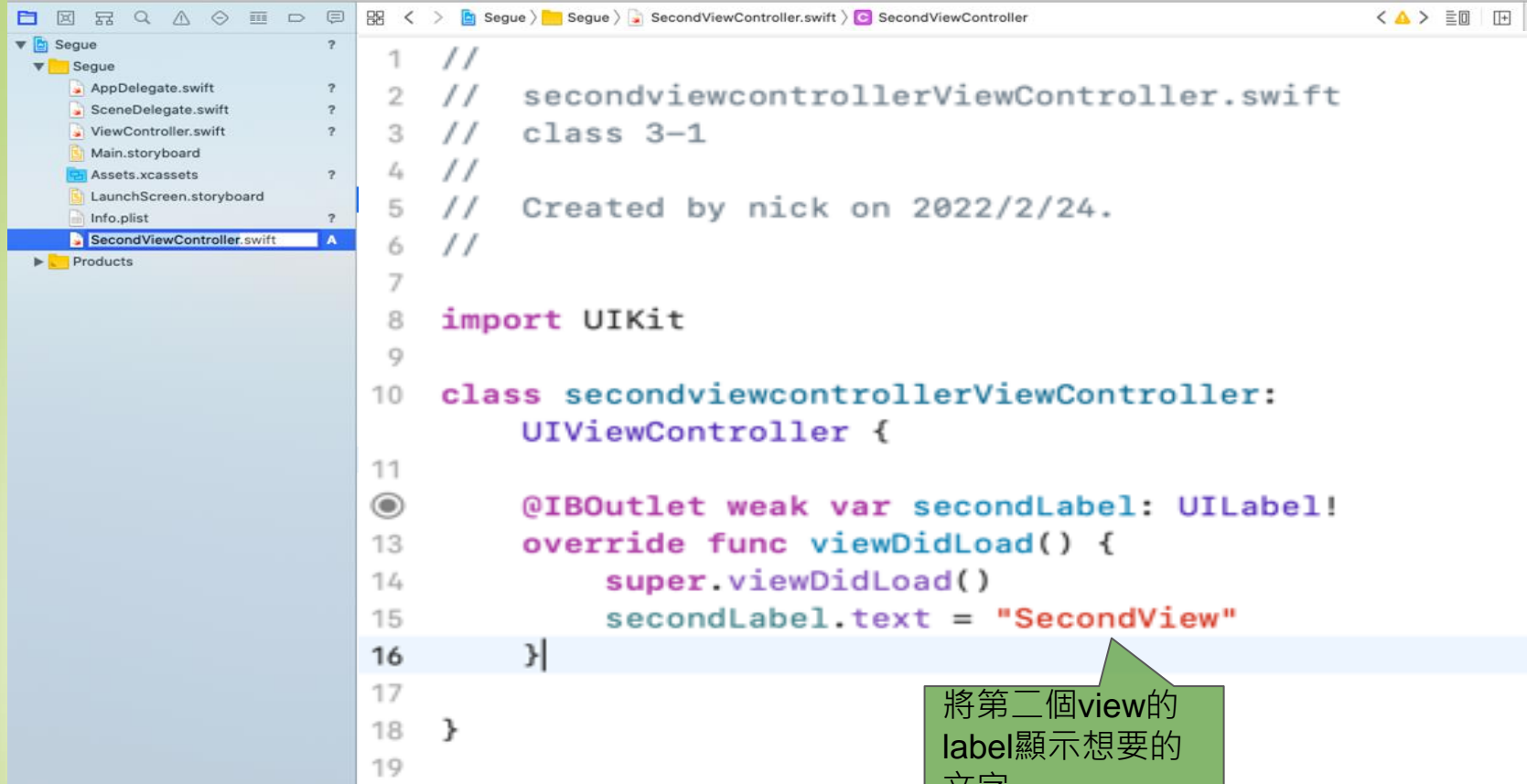
Next



# Segue



# Segue

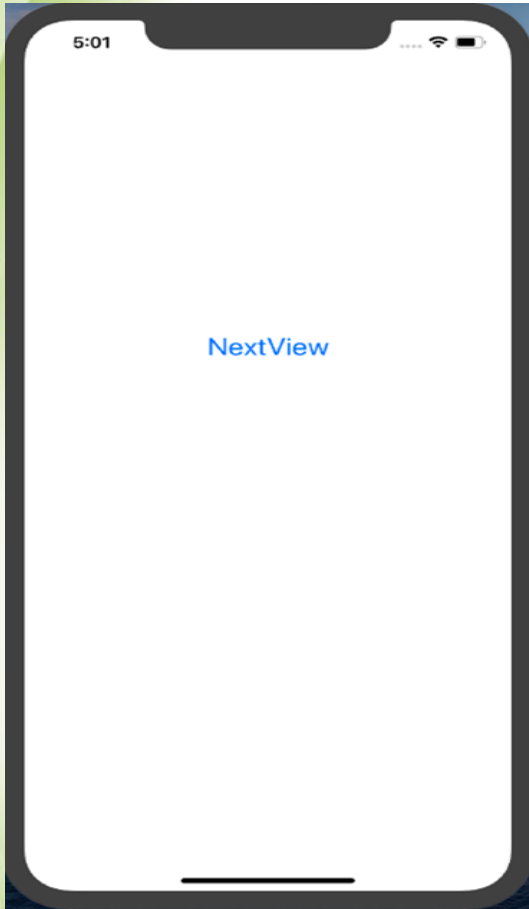


```
1  //
2  //  secondviewControllerViewController.swift
3  //  class 3-1
4  //
5  //  Created by nick on 2022/2/24.
6  //
7
8  import UIKit
9
10 class secondviewControllerViewController:
    UIViewController {
11
12     @IBOutlet weak var secondLabel: UILabel!
13     override func viewDidLoad() {
14         super.viewDidLoad()
15         secondLabel.text = "SecondView"
16     }
17
18 }
19
```

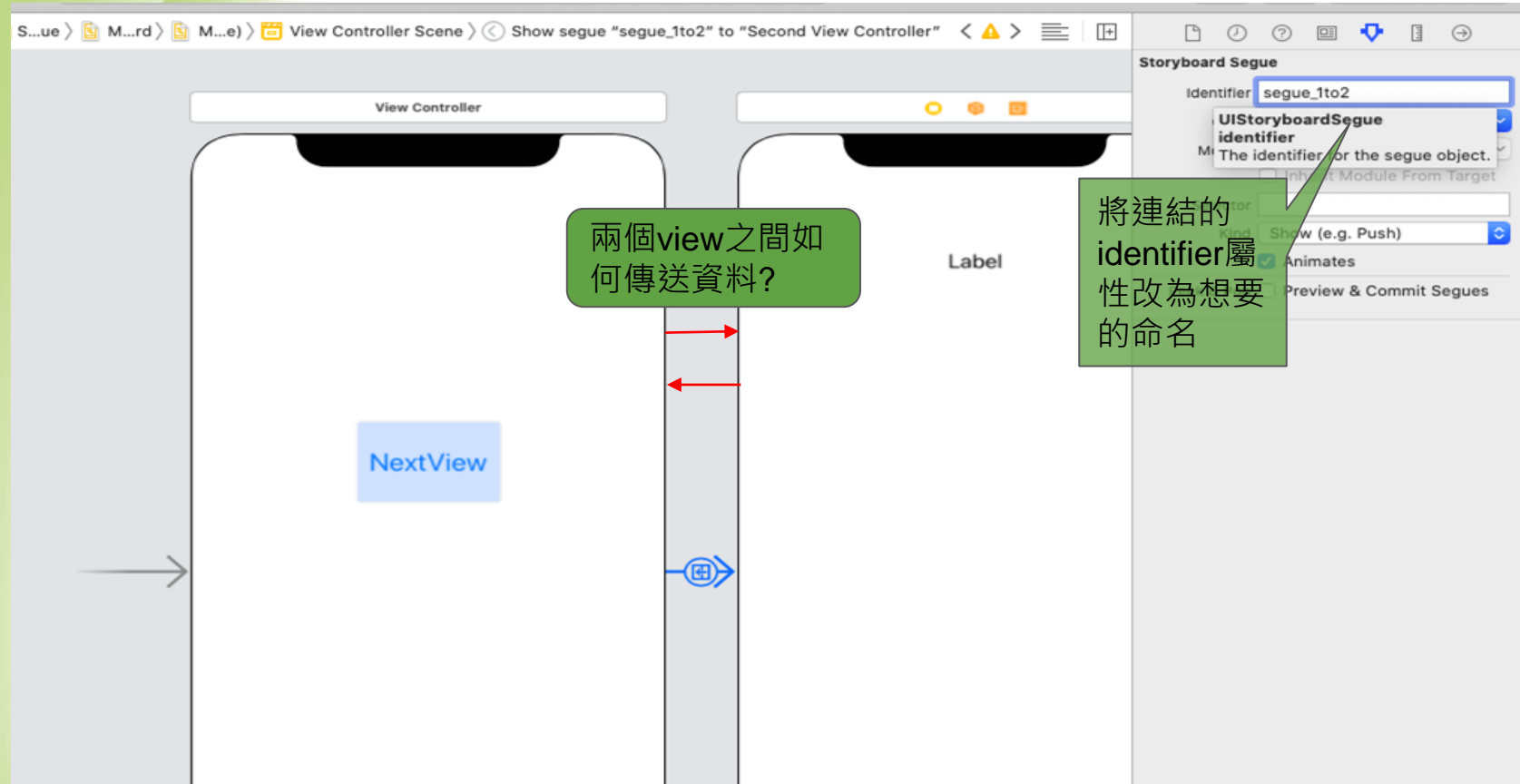
將第二個view的  
label顯示想要的  
文字

# Segue

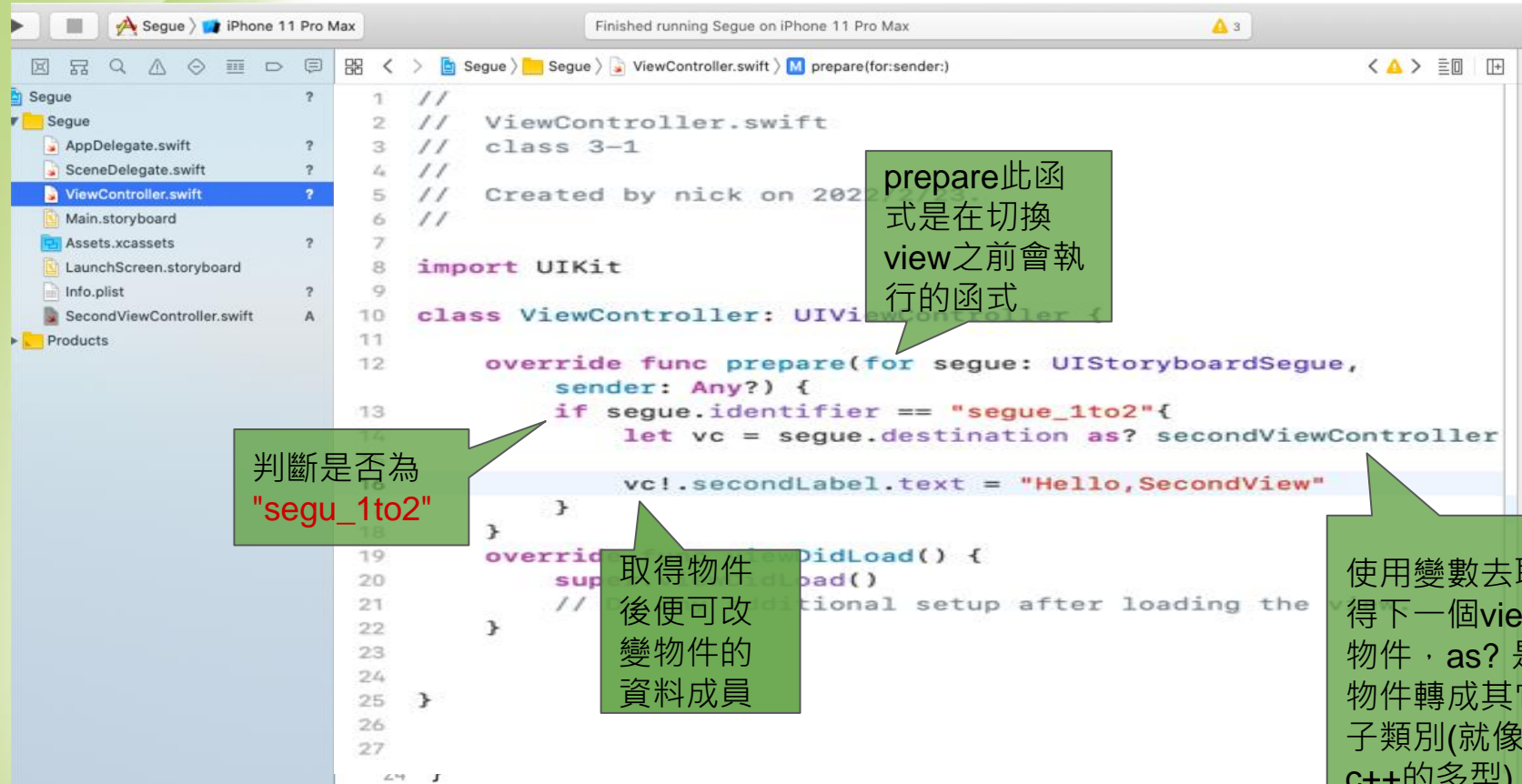
測試一下  
能否  
切換視窗



# Segue



# Segue



# Segue

```
Segue > Segue > SecondViewController.swift > SecondViewController

3
4
5
6
7
8
9 import UIKit
10
11 class SecondViewController: UIViewController {
12     var receiveStr: String? = nil
13     var sendBackStr: String? = nil
14
15     @IBOutlet weak var secondLabel: UILabel!
16     override func viewDidLoad() {
17         super.viewDidLoad()
18         if let str = receiveStr {
19             secondLabel.text = str
20         }
21         sendBackStr = "Back from sencond View"
22     }
23     /*
24     // MARK: - Navigation
25
```

宣告兩個字串一個是  
剛剛透過第一個view  
更新的字串，另一個  
是要讓第一個view讀  
取的字串

將更新過的字串  
顯示在Label

# Segue

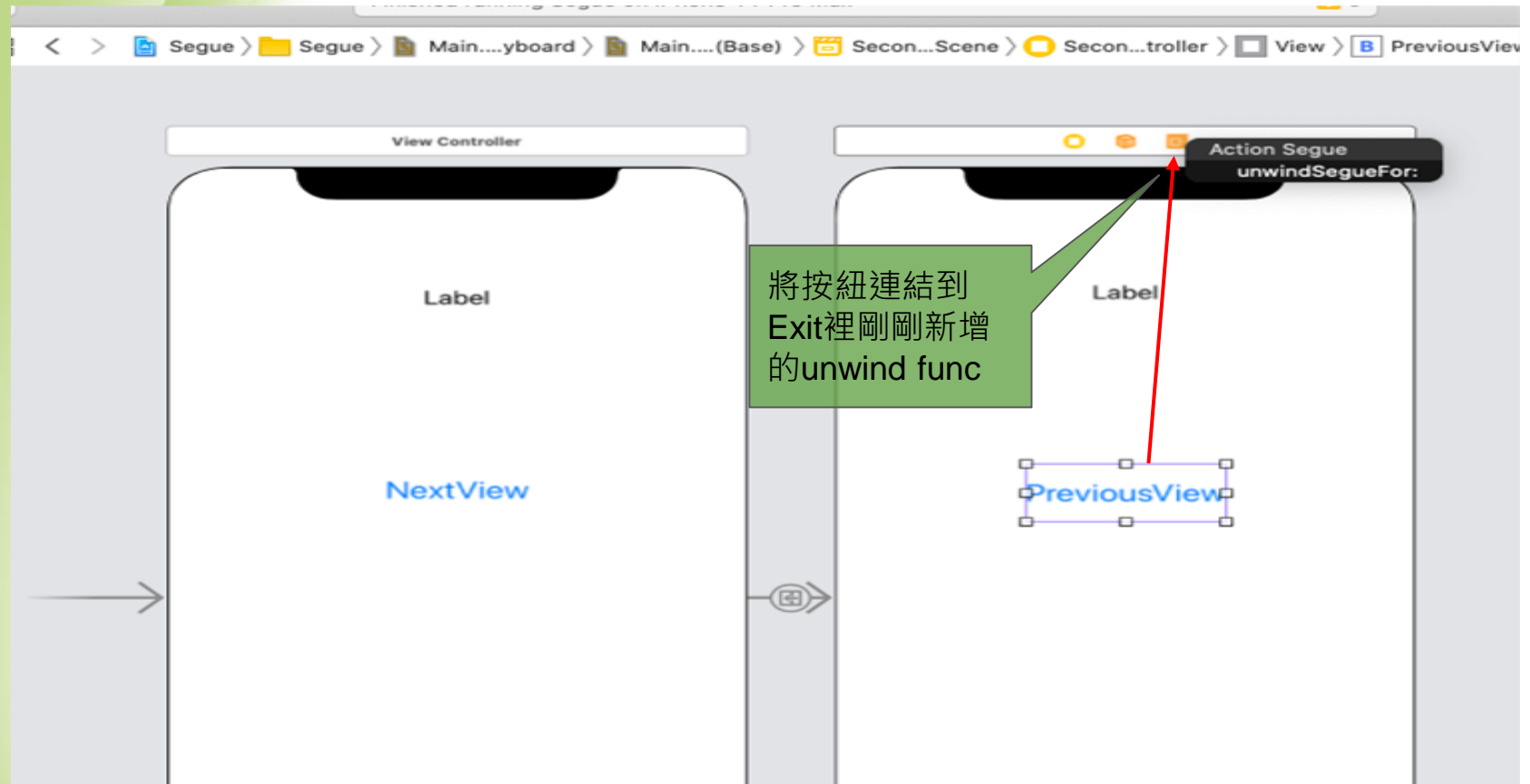
回來的方法  
使用unwind  
就好比在第一個view上  
做一個返回  
的記號

先將unwind  
的函式打好  
之後在與按  
鈕連結

這裡傳資  
料的方法  
大同小異

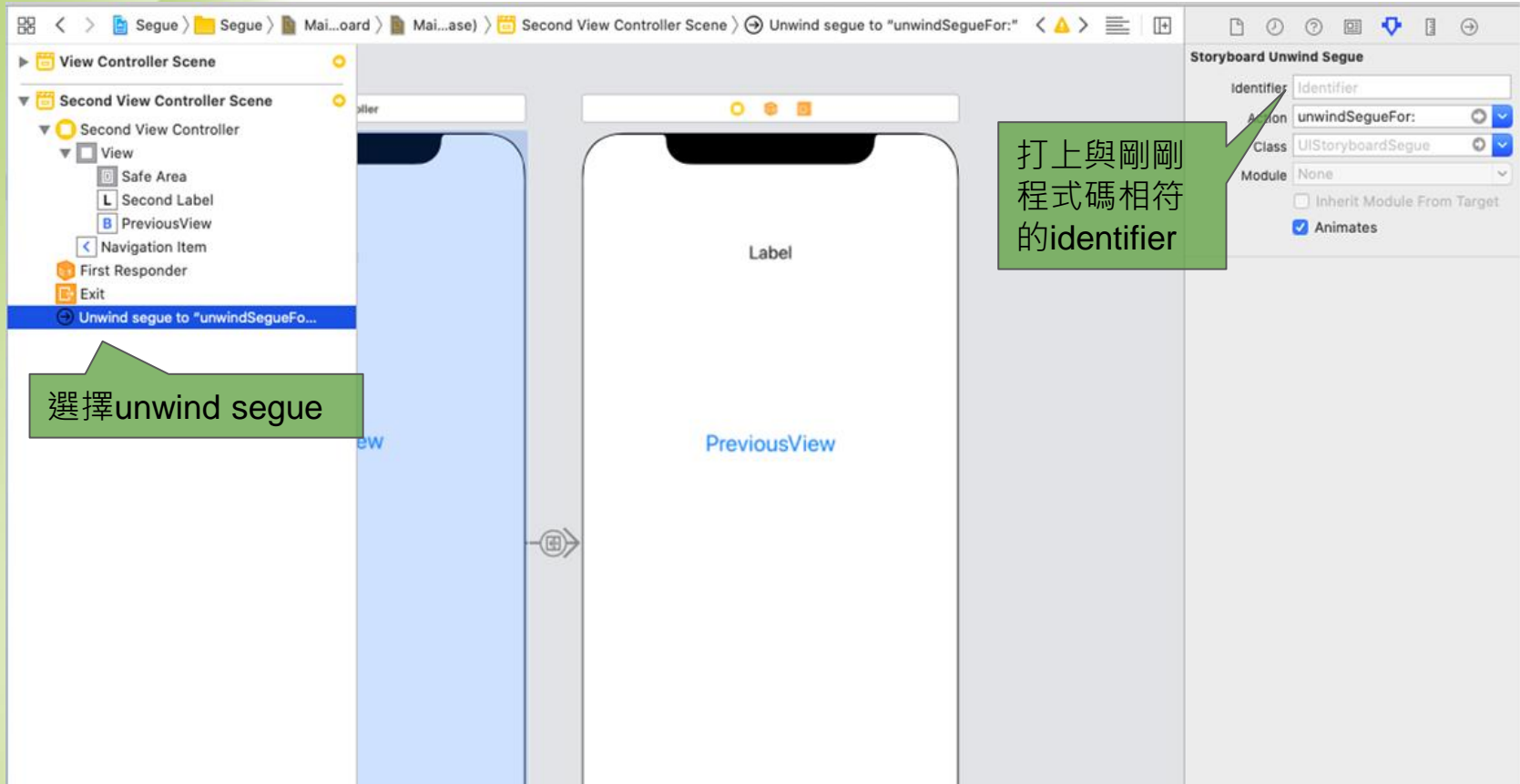
```
8
9  import UIKit
10
11  class ViewController: UIViewController {
12
13      @IBOutlet weak var firstLabel: UILabel!
14      @IBAction func unwindSegue(for segue: UIStoryboardSegue){
15          if segue.identifier == "unwind_segue"{
16              let vc = segue.source as! SecondViewController
17              if let str = vc.sendBackStr{
18                  firstLabel.text = str
19              }
20          }
21      }
22
23      override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
24          if segue.identifier == "unwind_segue"{
25              let vc = segue.destination as! SecondViewController
26              vc!.receiveStr = "Hello from SecondView"
27          }
28      }
29
30      override func viewDidLoad() {
31          super.viewDidLoad()
32          // Do any additional setup after loading the view.
33      }
34
35  }
```

# Segue

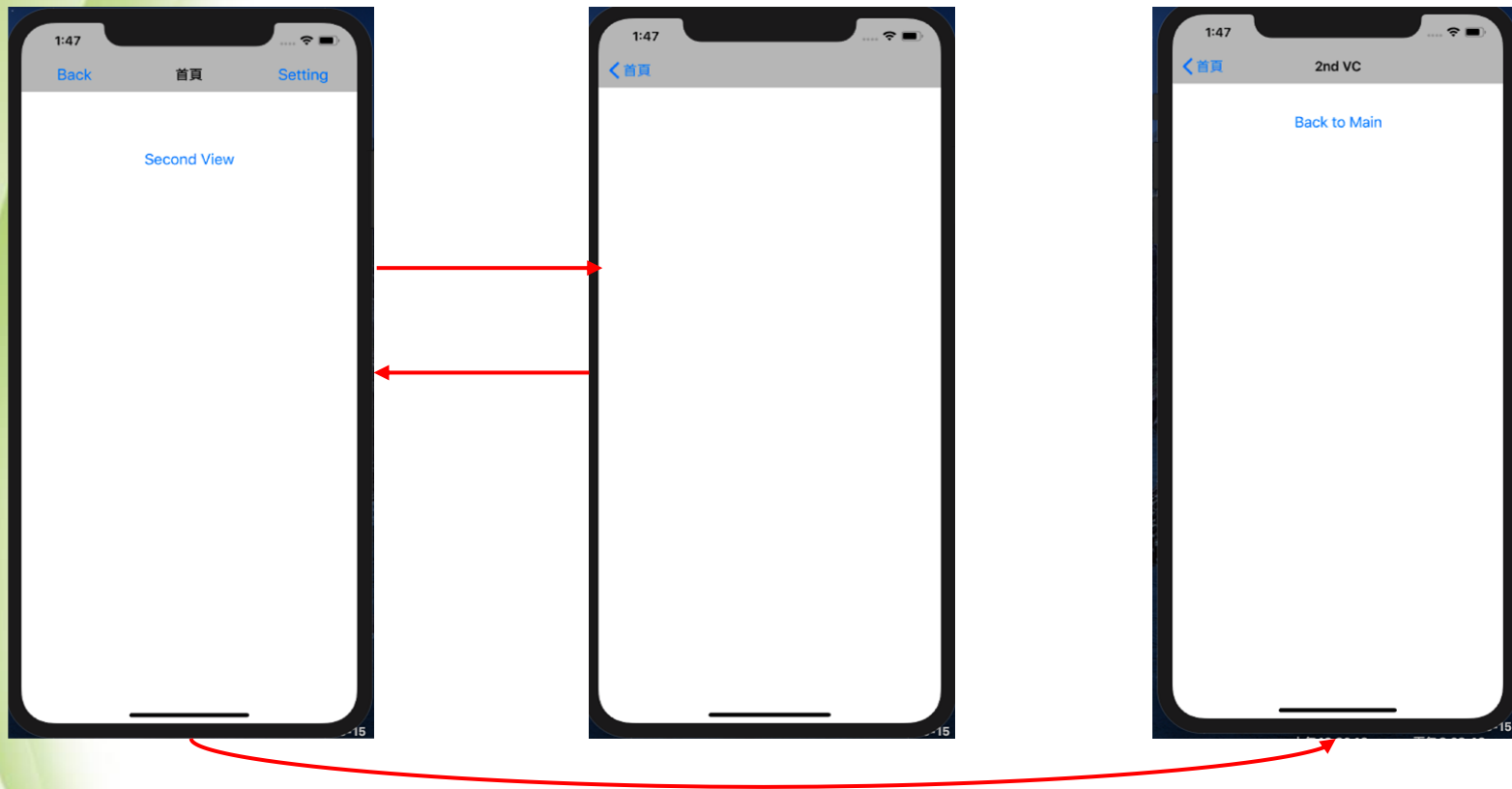




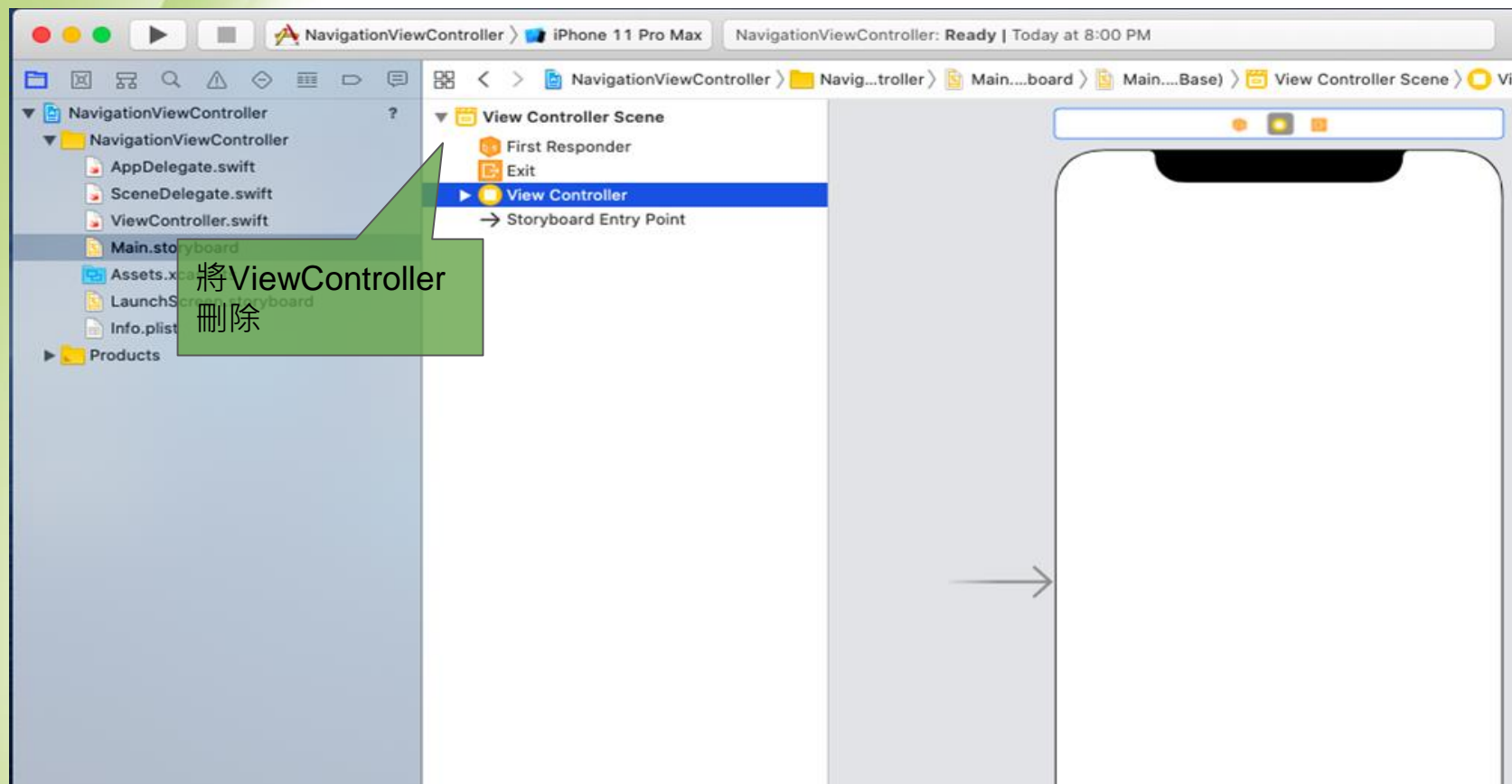
# Segue



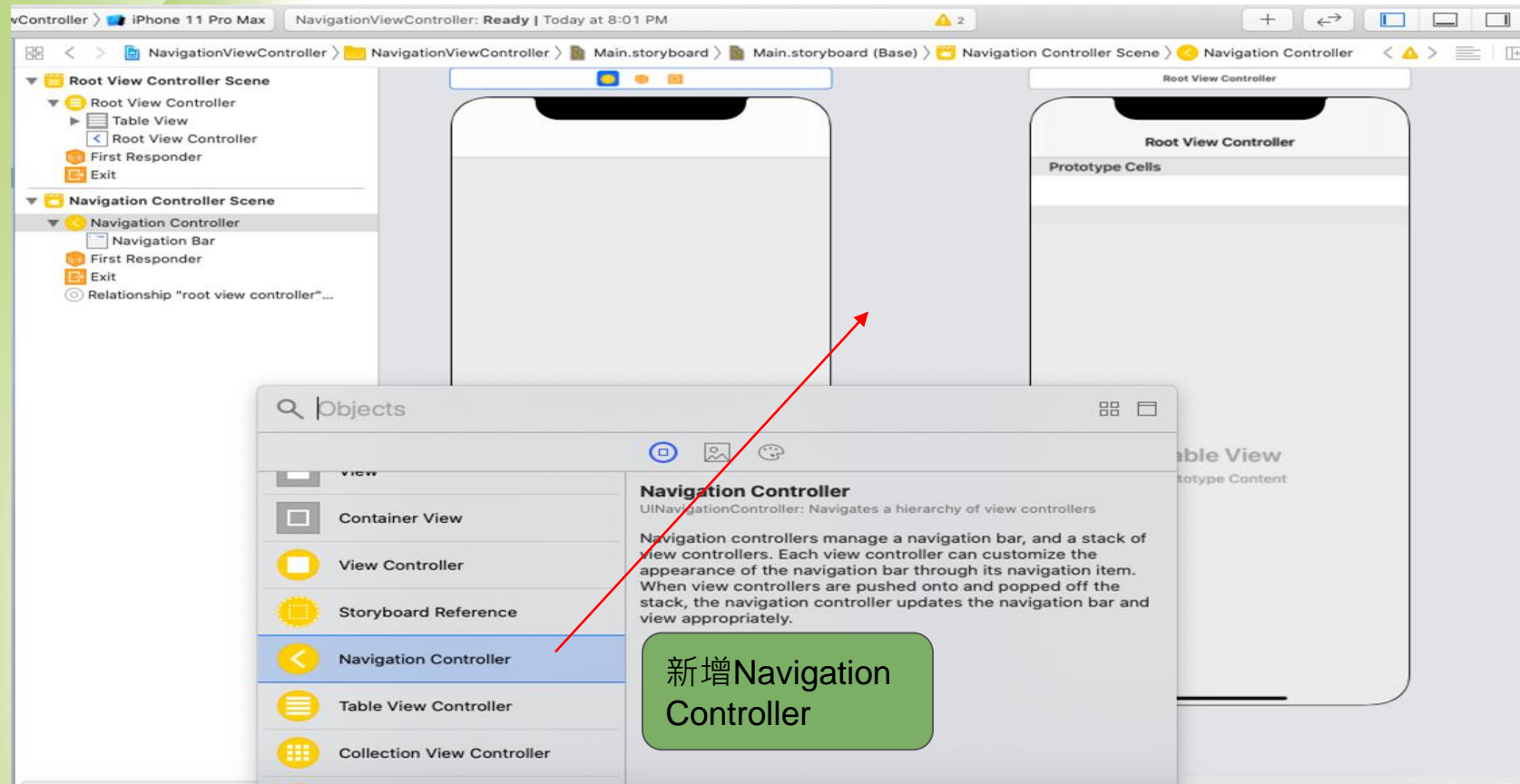
# NavigationViewController



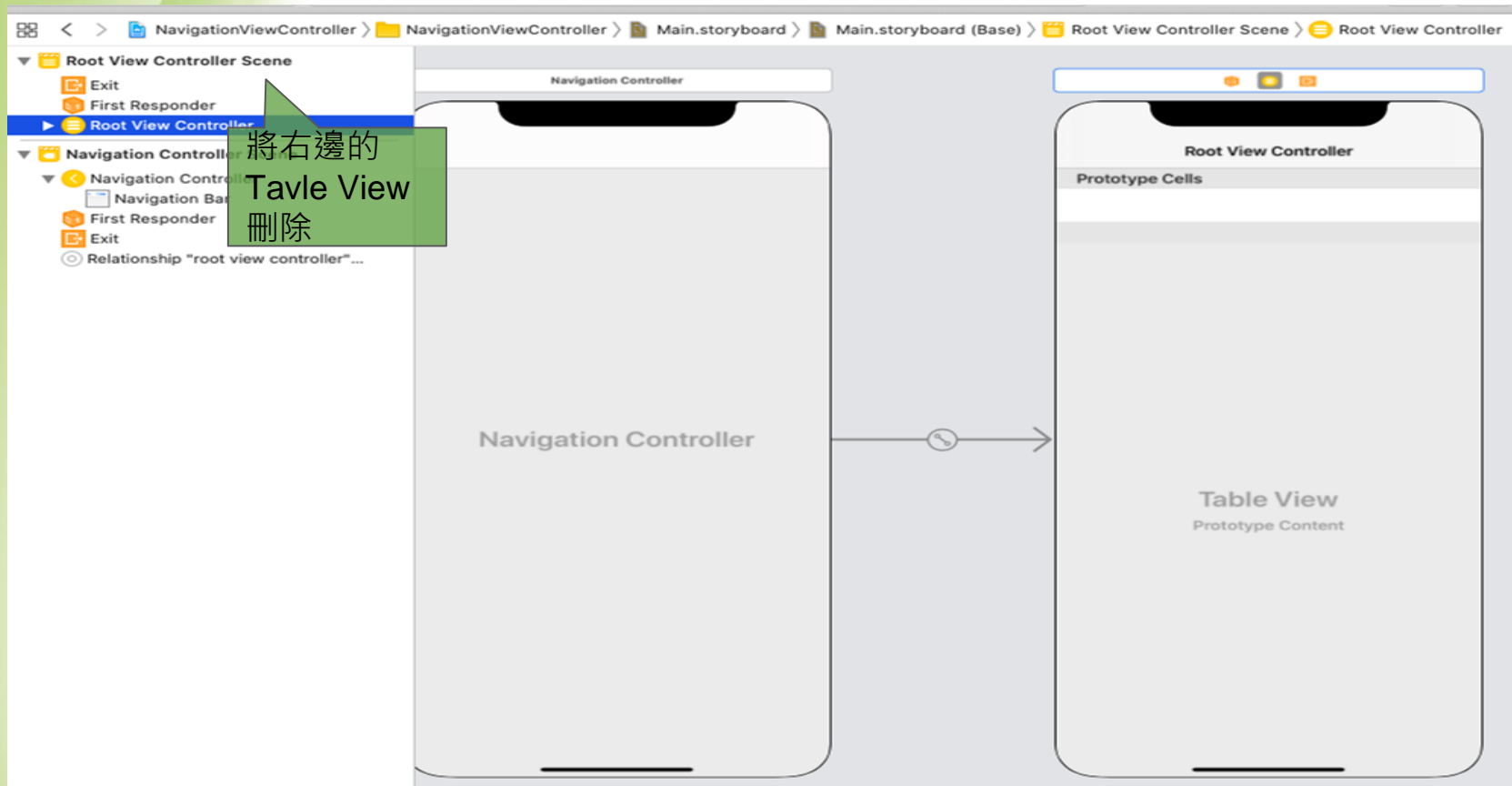
# NavigationViewController



# NavigationViewController



# NavigationViewController

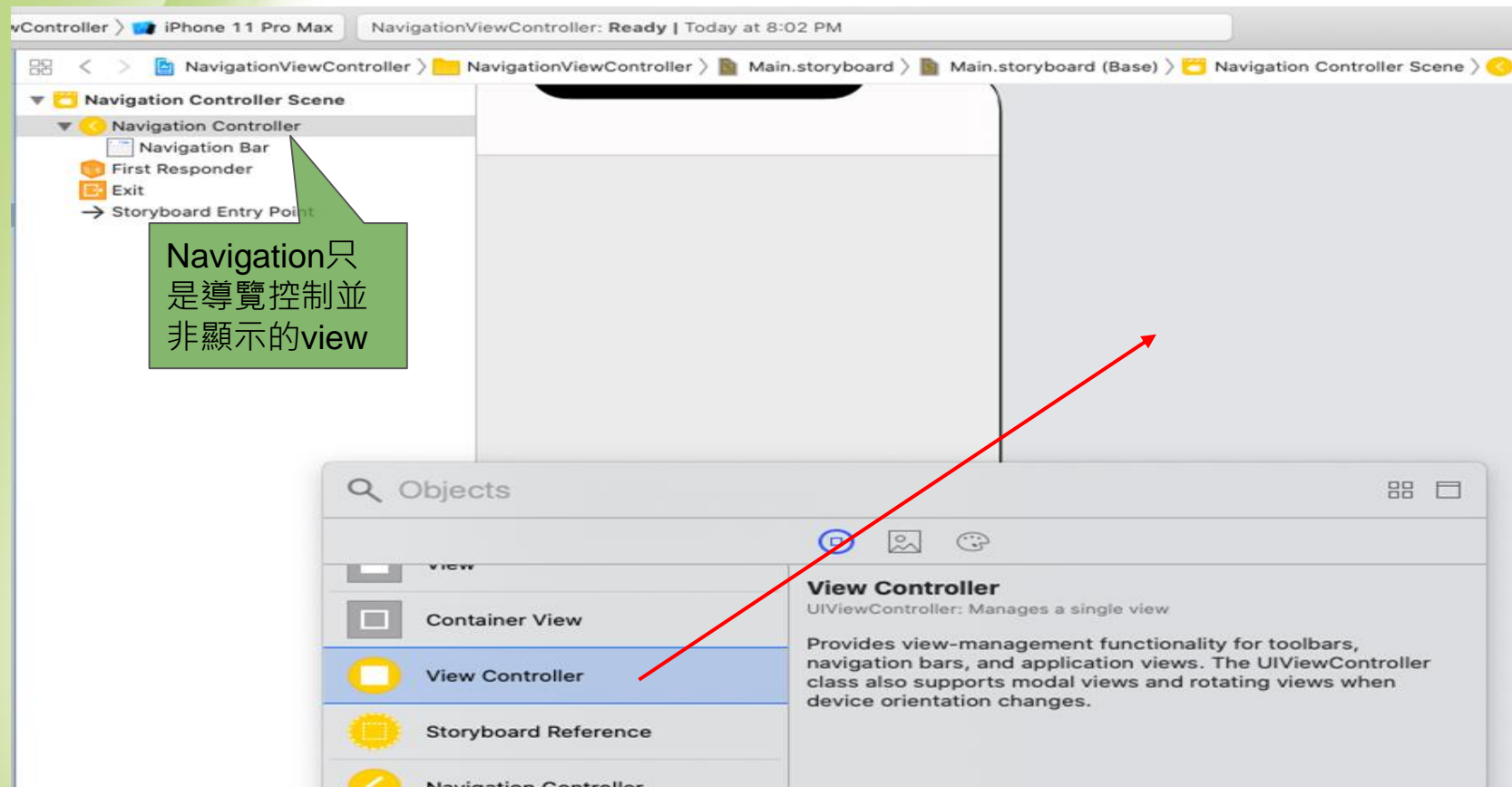


# NavigationViewController



設為程式  
開啟時的  
初始view

# NavigationViewController



# NavigationViewController





# NavigationViewController

The screenshot displays the Xcode interface for configuring a UINavigationController. The left sidebar shows the project hierarchy: UINavigationController Scene > Navigation Controller > Main Scene > Main > View Controller Scene > View Controller > View. The main canvas shows a UINavigationController containing a View Controller. A green callout bubble points to the 'Main' title in the View Controller's properties, with the text: 將第一個 view 的 Title 改為想要的名字 (Change the title of the first view to the name you want). Another green callout bubble points to the View Controller in the hierarchy, with the text: 將所有viewController新增完畢 (Finish adding all view controllers). The right sidebar shows the 'Simulated Metrics' and 'View Controller' settings. The 'View Controller' settings include: Title: Main, Is Initial View Controller: unchecked, Layout: Adjust Scroll View Insets (checked), Hide Bottom Bar on Push (unchecked), Resize View From NIB (checked), Use Full Screen (Deprecated) (unchecked), Extend Edges: Under Top Bars (checked), Under Bottom Bars (checked), Under Opaque Bars (unchecked), Transition Style: Cover Vertical, Presentation: Automatic, Defines Context (unchecked), Provides Context (unchecked), Content Size: Width 414, Height 896, Use Preferred Explicit Size (unchecked), Key Commands: Enter (Key), action (Selector).

將第一個 view 的 Title 改為想要的名字

將所有viewController新增完畢

# NavigationViewController

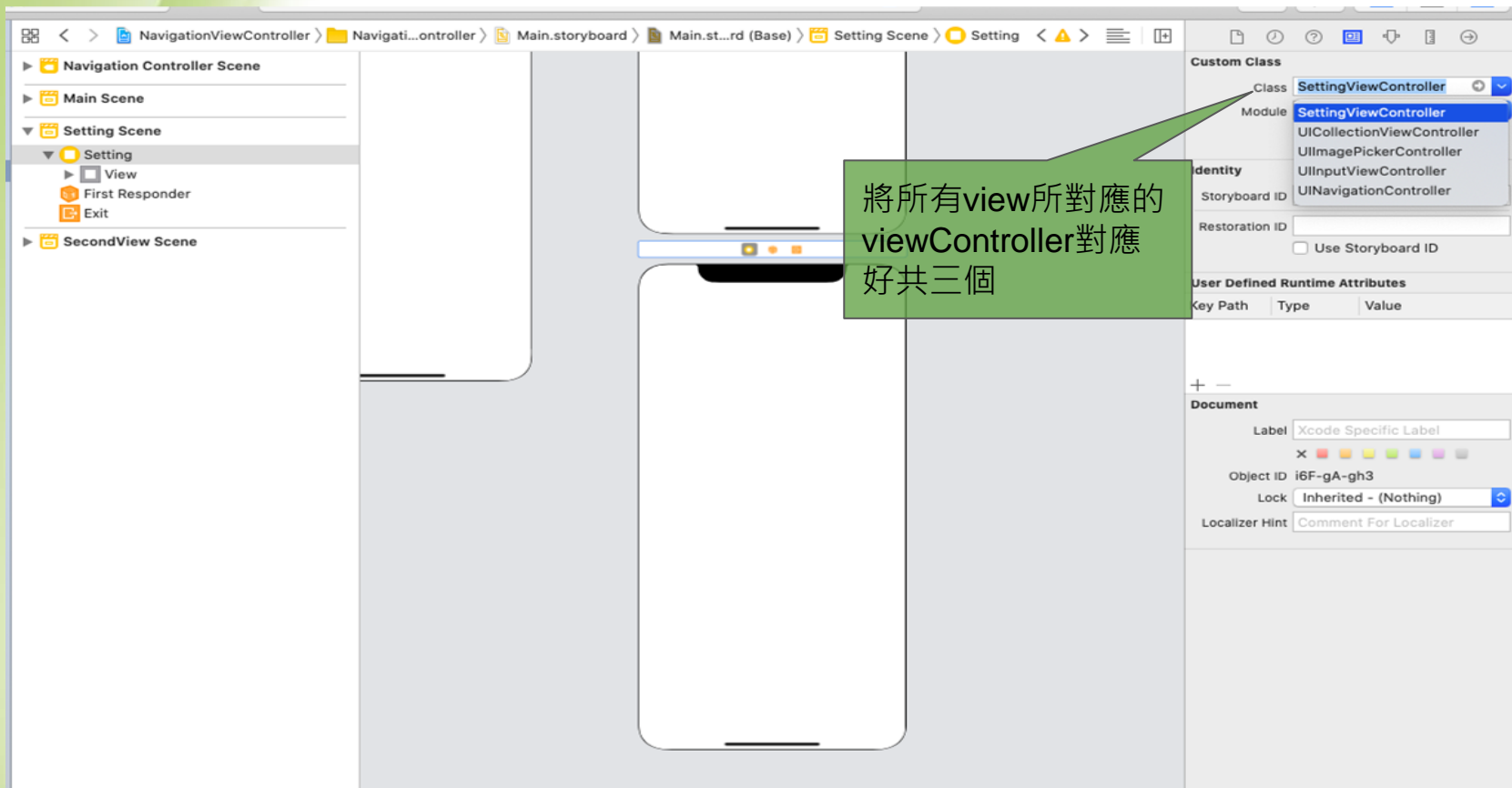


SecondViewController.swift

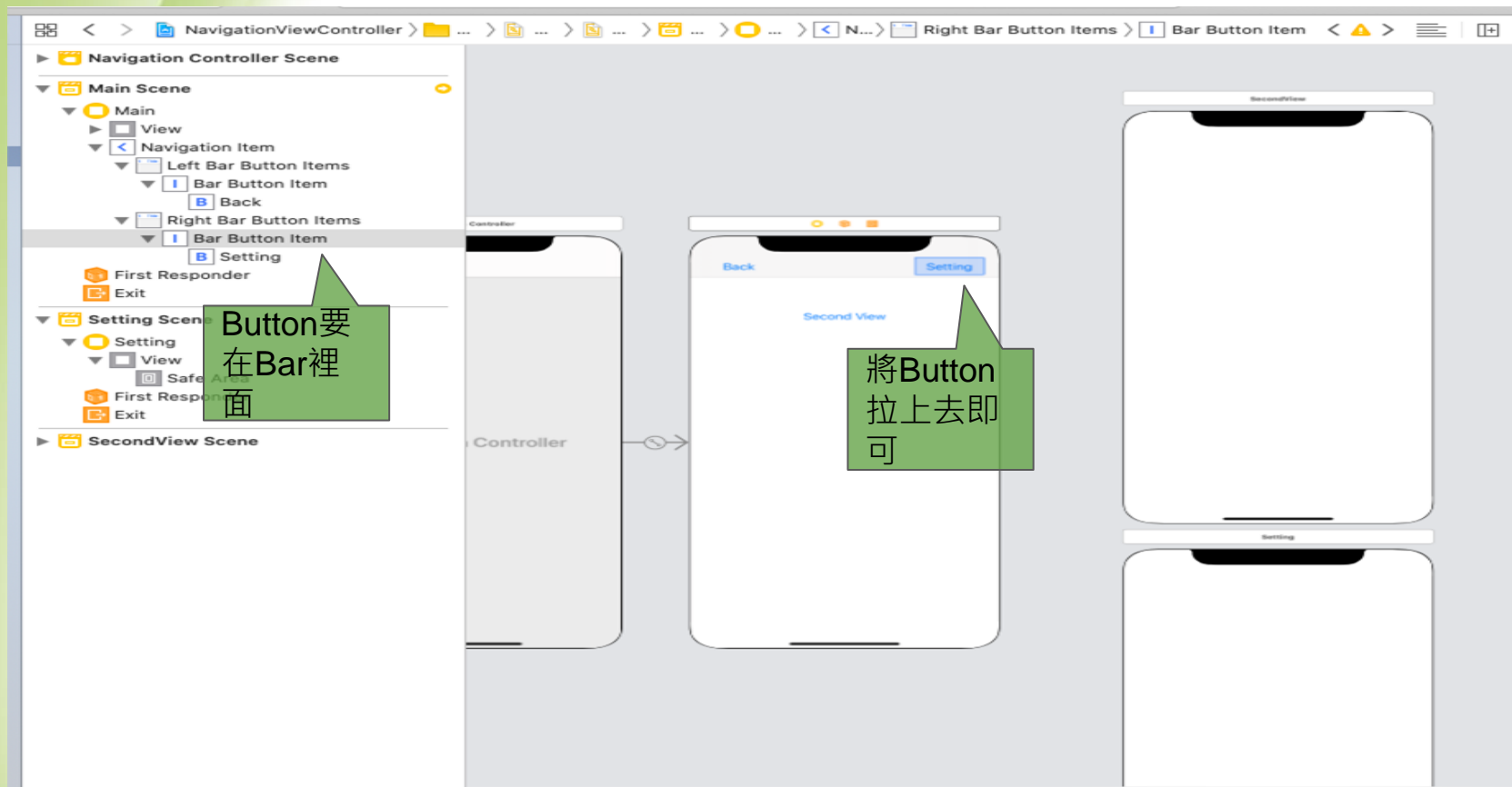


SettingViewController.swift

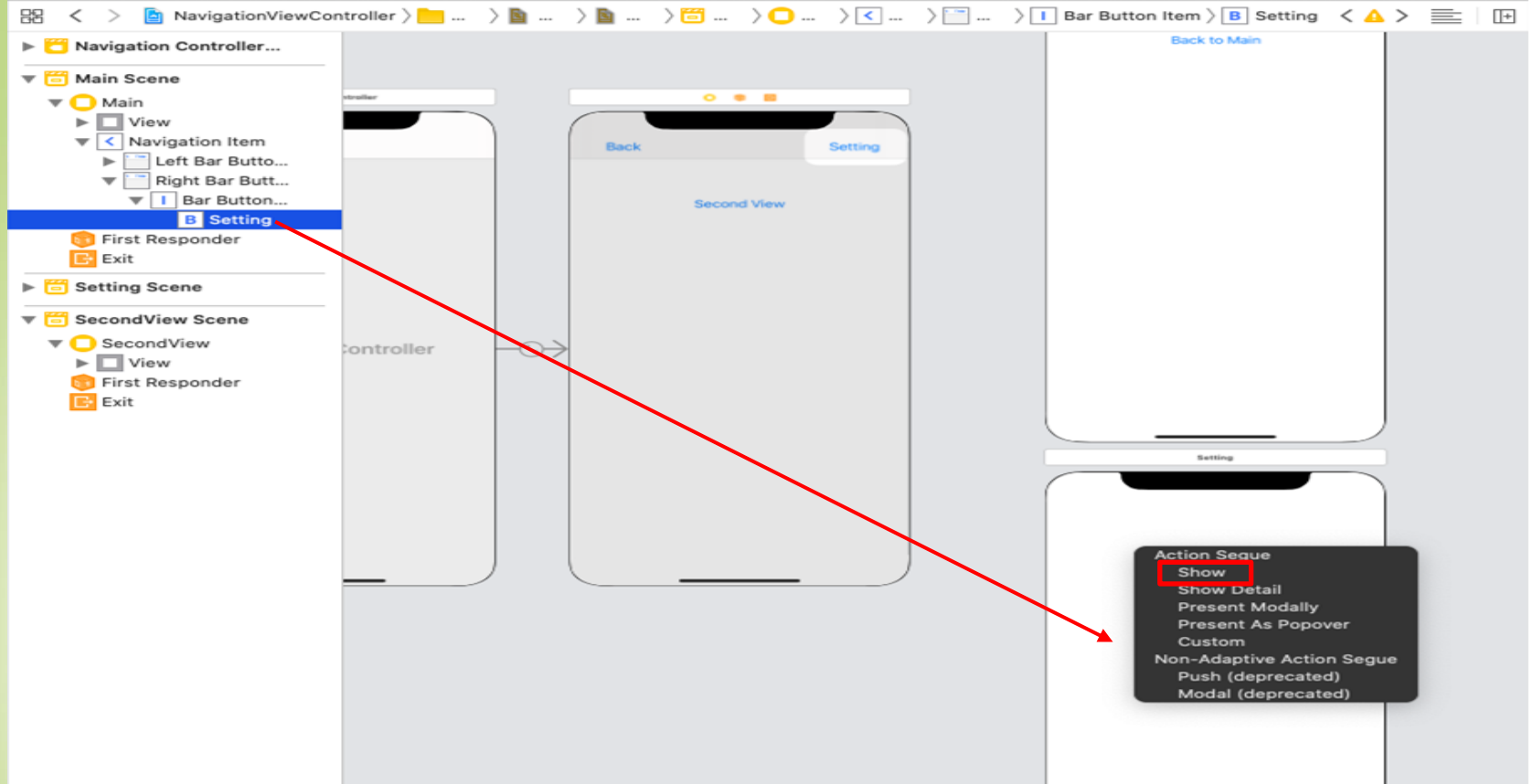
# NavigationViewController



# NavigationViewController



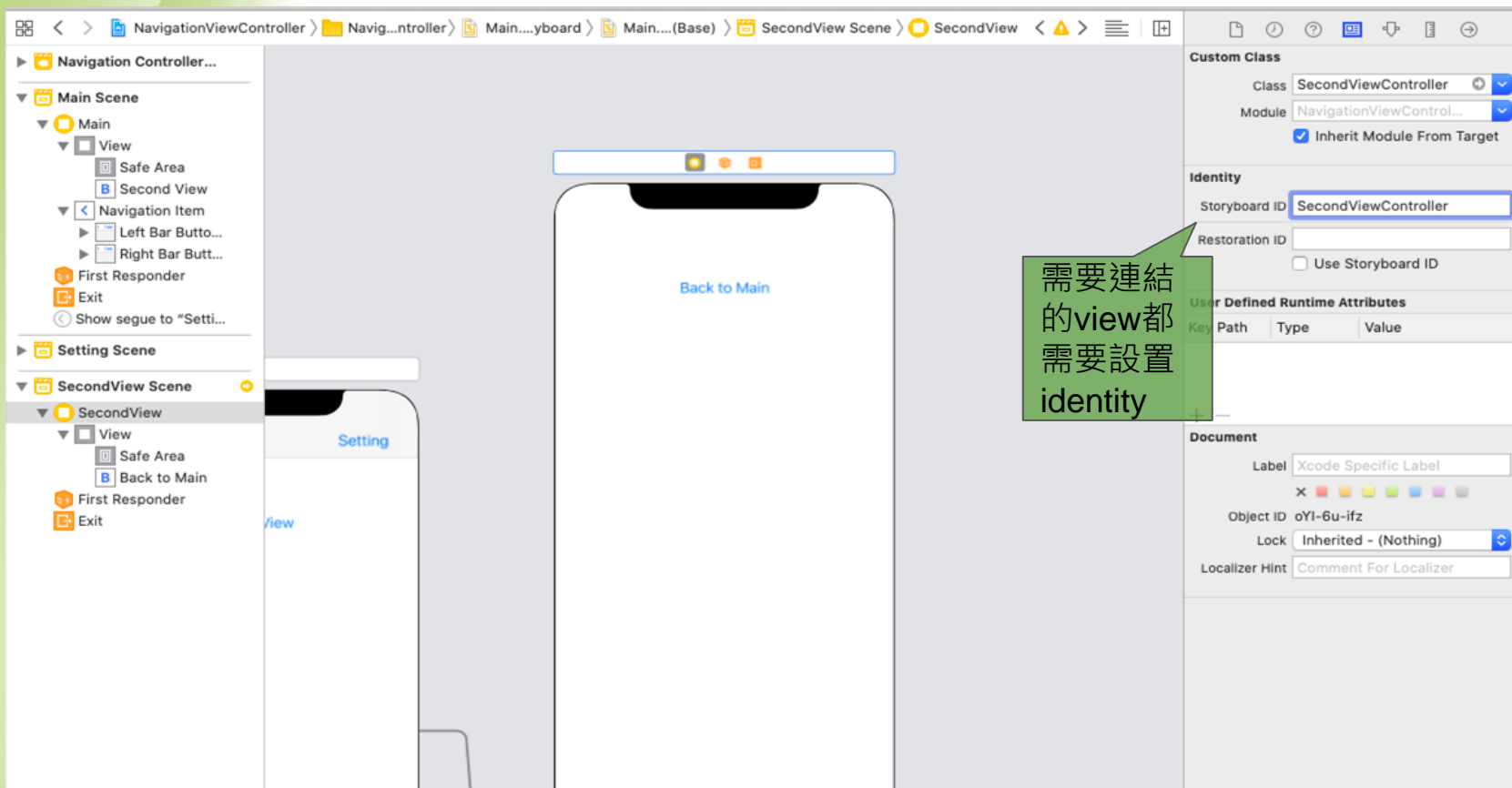
# NavigationViewController



# NavigationViewController



# NavigationViewController



# NavigationViewController

```
NavigationViewController > NavigationViewController > ViewController.swift > ViewController

2
3
4
5
6
7
8
9 import UIKit
10
11 class ViewController: UIViewController {
12
13     @IBAction func toSecondView(_ sender: UIButton) {
14         if let vc = storyboard?.instantiateViewController(withIdentifier:
            "SecondViewController"){
15             show(vc, sender: self)
16         }
17     }
18
19     override func viewDidLoad() {
20         super.viewDidLoad()
21         self.title = "首頁"
22         self.navigationController?.navigationBar.barTintColor =
            UIColor.lightGray
23         // Do any additional setup after loading the view.
24     }
25
26
27 }
```

新增按鈕第一  
一個view

顯示找到的view

利用剛剛設定  
過的識別名稱  
來找到要連結  
的view

將此view的  
標題更改並  
設定導覽欄  
的顏色



# NavigationViewController

```
NavigationViewController > NavigationViewController > SecondViewController.swift > M backToMain(:)
1
2
3
4
5
6
7
8
9 import UIKit
10
11 class SecondViewController: UIViewController {
12     @IBAction func backToMain(_ sender: UIButton) {
13         if let vc = storyboard?.instantiateViewController(withIdentifier:
            "ViewController"){
14             show(vc, sender: self)
15         }
16     }
17     override func viewDidLoad() {
18         super.viewDidLoad()
19         self.title = "2nd VC"
20
21         // Do any additional setup after loading the view.
22     }
23 }
```

製作返回  
的按鈕並  
連結

一樣使用識  
別的名字取  
得view