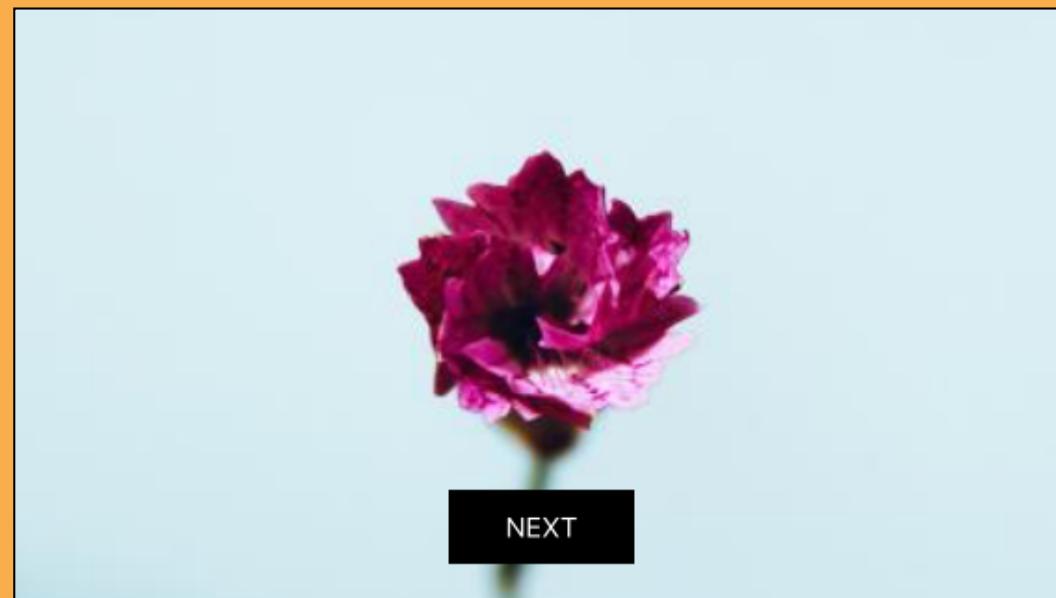




# **iPhone Development**

**GECF01 Workshop#7 - iOS Hands-on Lab**

**Adam**



# iPad



在中國，玫瑰則因其枝莖帶刺，被認為是刺客、俠客的象徵。而在西方則把玫瑰花當作嚴守秘密的象徵。做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳，於是有了Sub rosa，「在玫瑰花底下」這個拉丁成語。英語的under the rose則是源自德語unter der Rosen，古代德國的宴會廳、會議室以及酒店餐廳，天花板上常畫有或刻有玫瑰花，用來提醒與會者守口如瓶，嚴守秘密，不要把玫瑰花下的言行透露出去。這是起源於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事，她兒子丘比特為了幫他母親保有名節，於是給了他一朵玫瑰，請他守口如瓶，Horus收了玫瑰於是緘默不語，成為「沈默之神」，這就是under the rose之所以為守口如瓶的由來。



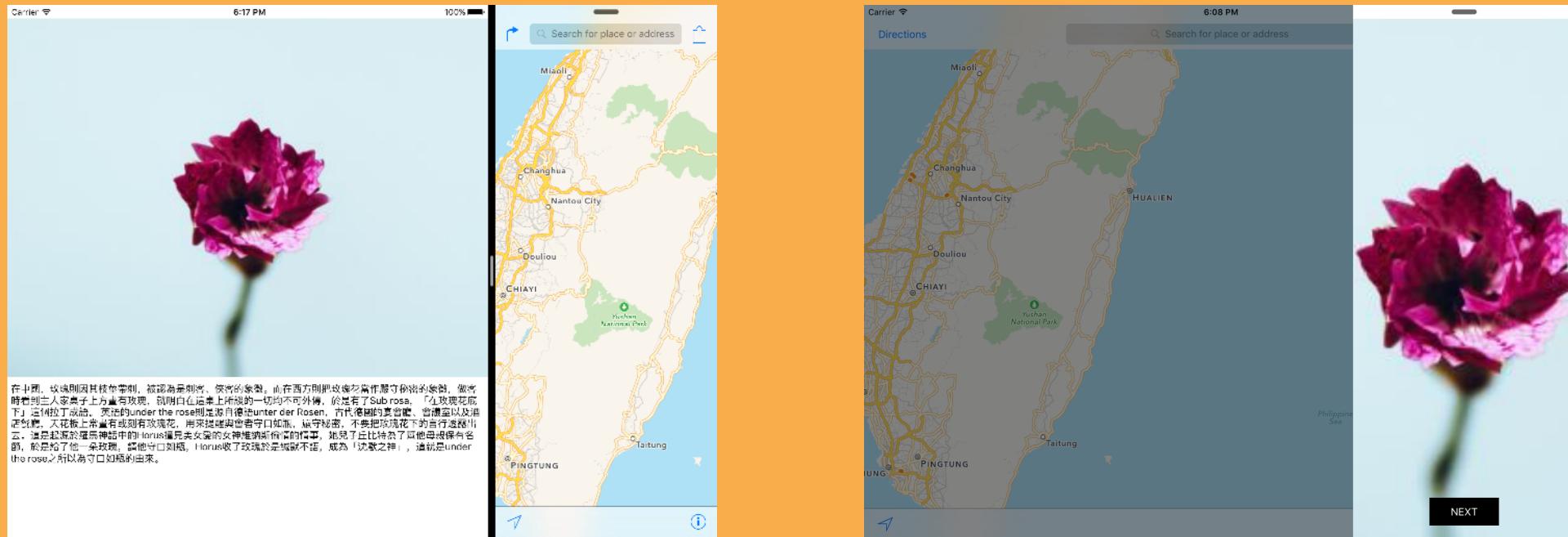
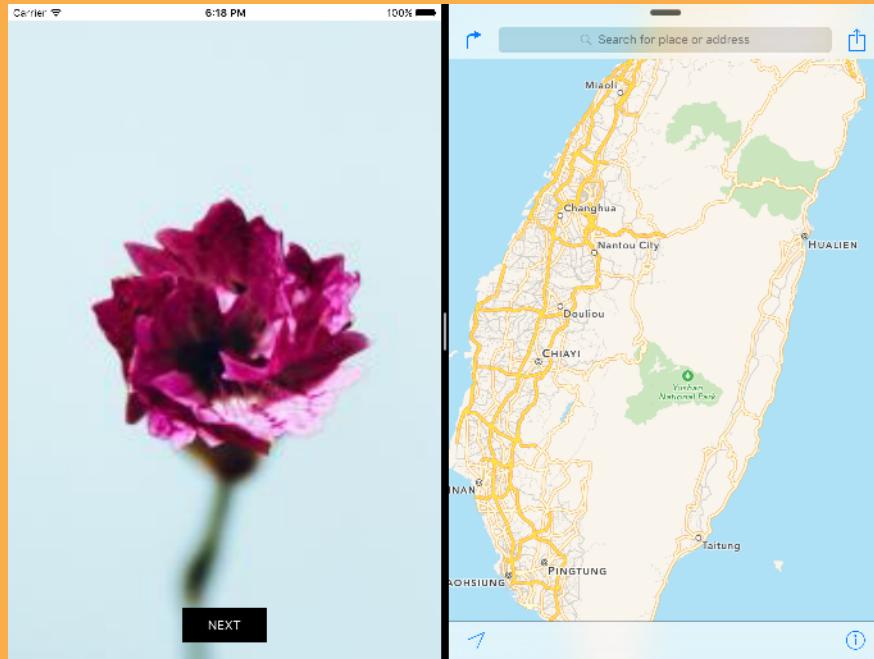
在中國，玫瑰則因其枝莖帶刺，被認為是刺客、俠客的象徵。而在西方則把玫瑰花當作嚴守秘密的象徵。做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳，於是有了Sub rosa，「在玫瑰花底下」這個拉丁成語。英語的under the rose則是源自德語unter der Rosen，古代德國的宴會廳、會議室以及酒店餐廳，天花板上常畫有或刻有玫瑰花，用來提醒與會者守口如瓶，嚴守秘密，不要把玫瑰花下的言行透露出去。這是起源於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事，她兒子丘比特為了幫他母親保有名節，於是給了他一朵玫瑰，請他守口如瓶，Horus收了玫瑰於是緘默不語，成為「沈默之神」，這就是under the rose之所以為守口如瓶的由來。

# iPhone 6 plus



在中國，玫瑰則因其枝莖帶刺，被認為是刺客、俠客的象徵。而在西方則把玫瑰花當作嚴守秘密的象徵，做客時看到主人家桌子上方還有玫瑰，就明白在這桌上所談的一切均不可外傳，於是有了Sub rosa，「在玫瑰花底下」這個拉丁成語。英語的under the rose則是源自德語unter der Rosen，古代德國的宴會廳、會議室以及酒店餐廳，天花板上常畫有或刻有玫瑰花，用來提醒與會者守口如瓶，嚴守秘密，不要把玫瑰花下的言行透露出去。這是起源於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事，她兒子丘比特為了幫他母親保有名節，於是給了他一朵玫瑰，請他守口如瓶，Horus收了玫瑰於是緘默不語，成為「沉默之神」，這就是under the rose之所以為守口如瓶的由來。

# iPad split mode



# iPad split mode





NEXT



NEXT



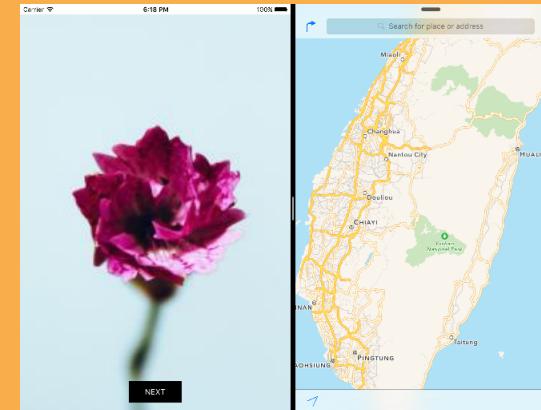
NEXT



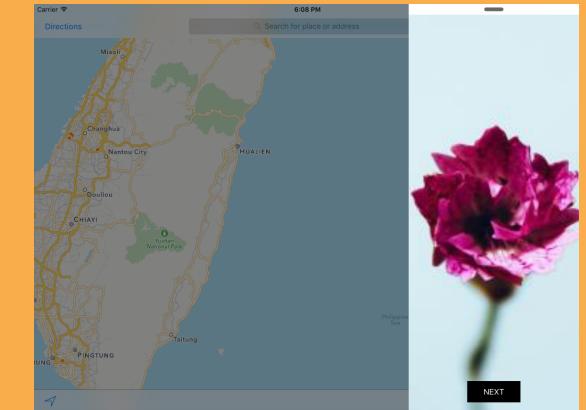
NEXT



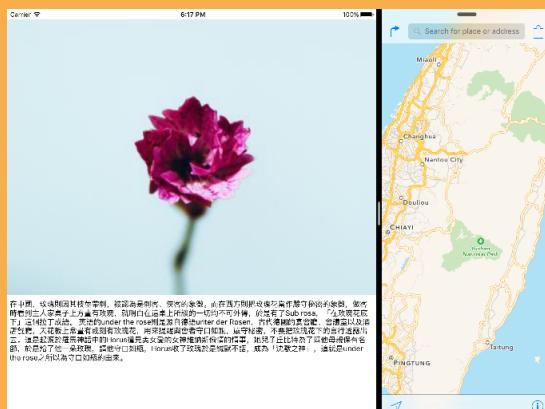
在中國，玫瑰則因其枝葉帶刺，被認為是暴利者、俠客的象徵。而在西方則把玫瑰花當作堅守秘密的象徵。做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳。於是有了Sub rosa，在玫瑰花底下。這個詞源來自於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事。她兒子丘比特為了幫他母親保住名節，於是給了他一朵玫瑰。請他守口如瓶。Horus收了玫瑰於是沈默不語，成為「沉默之神」。這就是under the rose之所以為守口如瓶的由來。



NEXT



NEXT



在中國，玫瑰則因其枝葉帶刺，被認為是暴利者、俠客的象徵。而在西方則把玫瑰花當作堅守秘密的象徵。做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳。於是有了Sub rosa，在玫瑰花底下。這個詞源來自於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事。她兒子丘比特為了幫他母親保住名節，於是給了他一朵玫瑰。請他守口如瓶。Horus收了玫瑰於是沈默不語，成為「沉默之神」。這就是under the rose之所以為守口如瓶的由來。



NEXT

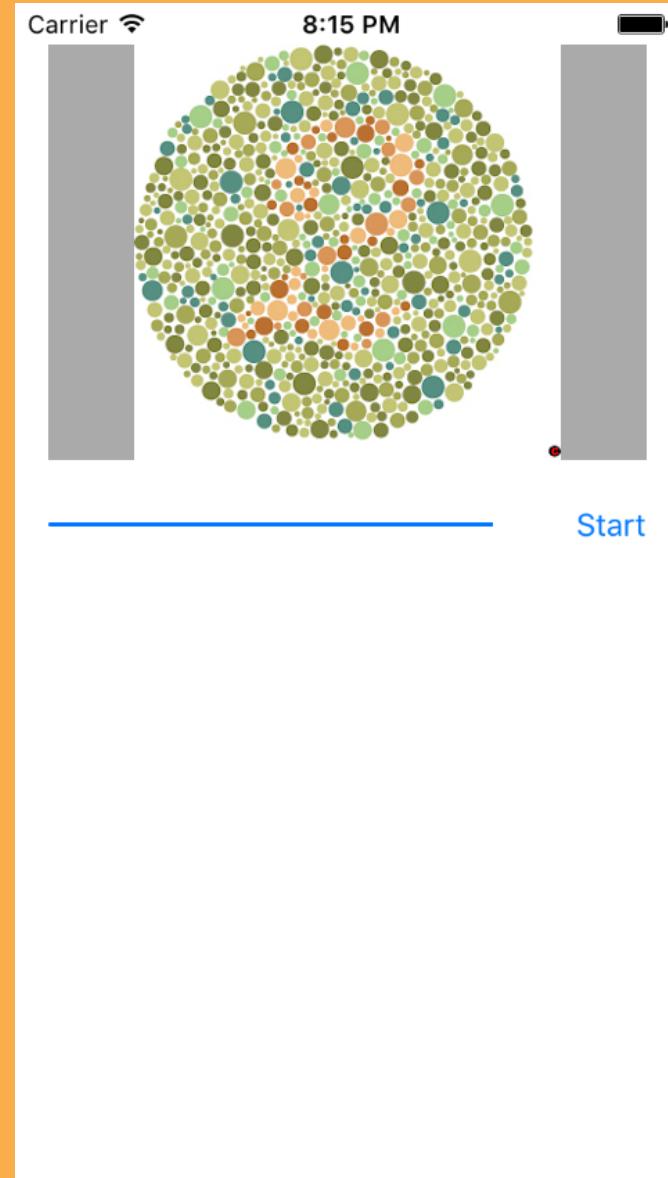
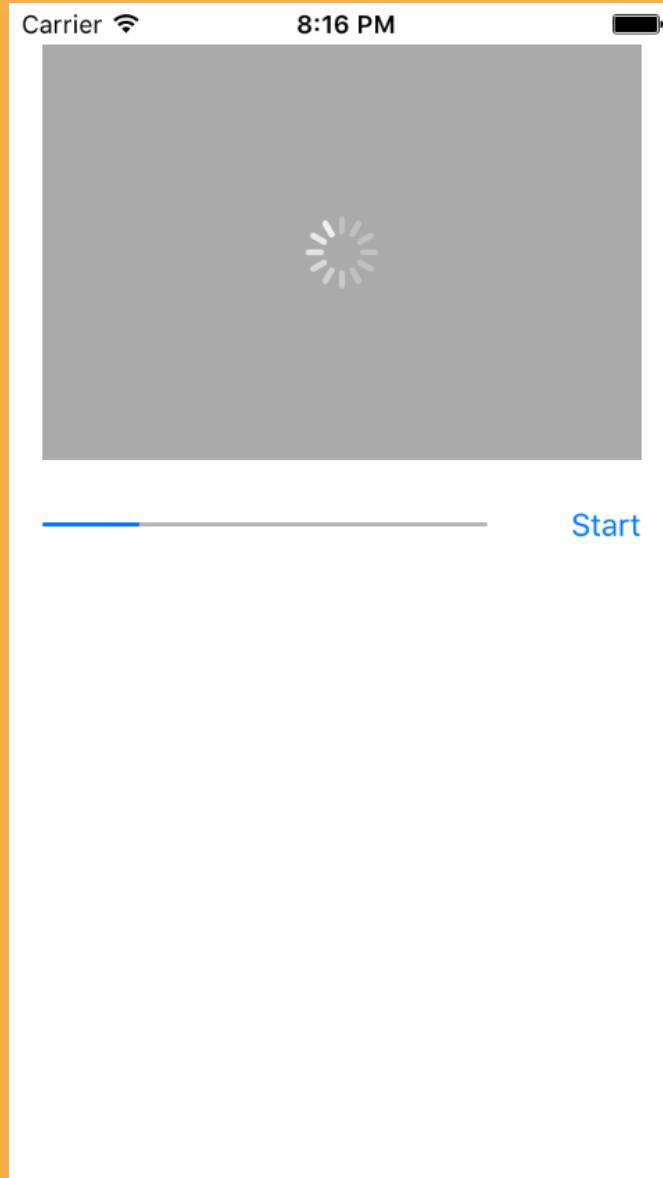


NEXT



在中國，玫瑰則因其枝葉帶刺，被認為是暴利者、俠客的象徵。而在西方則把玫瑰花當作堅守秘密的象徵。做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳。於是有了Sub rosa，在玫瑰花底下。這個詞源來自於羅馬神話中的Horus撞見美女愛的女神維納斯偷情的情事。她兒子丘比特為了幫他母親保住名節，於是給了他一朵玫瑰。請他守口如瓶。Horus收了玫瑰於是沈默不語，成為「沉默之神」。這就是under the rose之所以為守口如瓶的由來。

# Download image sample



# Data request sample



登入平臺

會員服務

平臺關鍵字

查詢

資料集下載

互動專區

活化應用

諮詢小組

M2M專區

授權條款

關於平臺

相關連結

## Open Data 顯微鏡

讓政府開放資料帶你細看醫療問題!



<http://data.gov.tw>

# 原鄉旅遊路線

資料集評分:   
平均: 4 (2 投票)

[訂閱](#) [訂閱說明](#)

資料集描述	原住民族地區之旅遊規劃路線之參考。
主要欄位說明	序號、資料日期、路線名稱、民族、所在地、交通方式、行程內容、旅遊季節、旅遊天數、行程特色、部落資訊(負責機構)、重要祭典、問候語、成行人數、每人費用、住宿地點、組合。
資料資源	<a href="#"> XML</a> <a href="#"> 檢視資料</a> <a href="#"> JSON</a> <a href="#"> 檢視資料</a> <a href="#"> CSV</a> <a href="#"> 檢視資料</a>
資料集類型	原始資料
資料集提供機關名稱	原住民族委員會

<http://data.gov.tw/node/7128>

```
{  
    "Seq": 1,  
    "DateListed": "2013/09/01",  
    "Name": "老地方展新魅力—台東部落逍遙遊",  
    "Ethnic": "魯凱族、排灣族",  
    "Region": "臺東縣金峰鄉、太麻里鄉、卑南鄉、臺東市",  
    "Access": "",  
    "TravelItinerary": "一日遊：屏東縣→達仁鄉→大武鄉→多良向陽  
    "TouristSeason": "",  
    "TravelDays": "一個太陽",  
    "TravelCharacteristics": "多良火車站 / 多良向陽薪傳木工坊",  
    "TribeInformation": "",  
    "ImportantRite": "",  
    "Greetings": "",  
    "AssociationPopulation": "",  
    "CostPerPerson": "",  
    "Accommodation": "",  
    "Group": "2013年60條原住民族深度旅遊路線"  
},  
{  
    "Seq": 2,  
    "DateListed": "2013/09/01",  
    "Name": "花東縱谷單車人文旅行—鄉村、溫泉、部落",  
    "Ethnic": "阿美族、撒奇萊雅族、布農族、卑南族",  
    "Region": "花蓮縣、台東縣",  
    "Access": "",  
    "TravelItinerary": "第一天：花蓮火車站→德固灣撒奇萊雅文化廣場→...  
    "TouristSeason": "",  
    "TravelDays": "三個太陽二個月亮",  
    "TravelCharacteristics": "花東縱谷線 如德固灣撒奇萊雅文化廣場",  
    "TribeInformation": "",  
    "ImportantRite": "",  
    "Greetings": "",  
    "AssociationPopulation": "",  
    "CostPerPerson": "",  
    "Accommodation": "",  
    "Group": "2013年60條原住民族深度旅遊路線"  
},
```

Carrier 7:41 PM

# 原鄉旅遊路線

**老地方展新魅力—台東部落逍遙遊**  
一日遊：屏東縣→達仁鄉→大武鄉→多良向陽薪傳木工坊→金崙溫泉...

**花東縱谷單車人文旅行-鄉村、溫泉、部落**  
第一天：花蓮火車站→德固灣撒奇萊雅文化廣場、撒奇萊雅酒莊→吉...

**阿美族巴歌浪舞動之夜~水梯田海稻米、米粑流...**  
第一天：花蓮火車站→箭瑛大橋(1977年兩位老師的故事)→海岸山脈...

**南臺灣原鄉部落騎遇記(挑戰級)**  
第一天：高鐵左營站專車接送美濃人字山莊→109線道美濃自行車道...

**茂林紫斑蝶禮納里部落體驗二日遊**  
第一天：台北火車站→禮納里原住民風味餐→禮納里好茶部落 (好茶...

**苗栗泰雅原鄉產業一日遊**  
一日遊：苗栗泰安鄉→司馬限市集→麥路豐市集→麻必浩市集→士林...

**烏來福山部落悠遊行~溪流德拉楠. 巧遇馬告香**  
一日遊：烏來出發 → 五重溪瀑布 → 水圳步道散步趣 → 蝴蝶公園→部...

**探尋祕境禮納里部落二日遊**  
第一天：台北火車站→(中餐)禮納里風味餐→禮納里部落專人導覽→...

**渡邊吉海秧~大農大富平地森林單車遊**  
第一天：花蓮火車站→馬太鞍溪無毒農場→大農大富平地森林單車遊...

**嘎色鬧泰雅農夫~竹頭角木琴樂舞一日遊**  
一日遊：桃園縣復興鄉→桃園縣原住民文化會館→樂信瓦旦紀念公園...

**學習土地倫理~巒山森林博物館**  
第一天：海端鄉(布農族文物館)→拉瑪達星星雕塑像(布農族抗日英雄...

**來去烏來住一晚~文化. 遇見繽紛泰雅(二日遊)**  
第一天：新店捷運站相見歡→內洞國家森林遊樂區→泰雅傳統美食→...

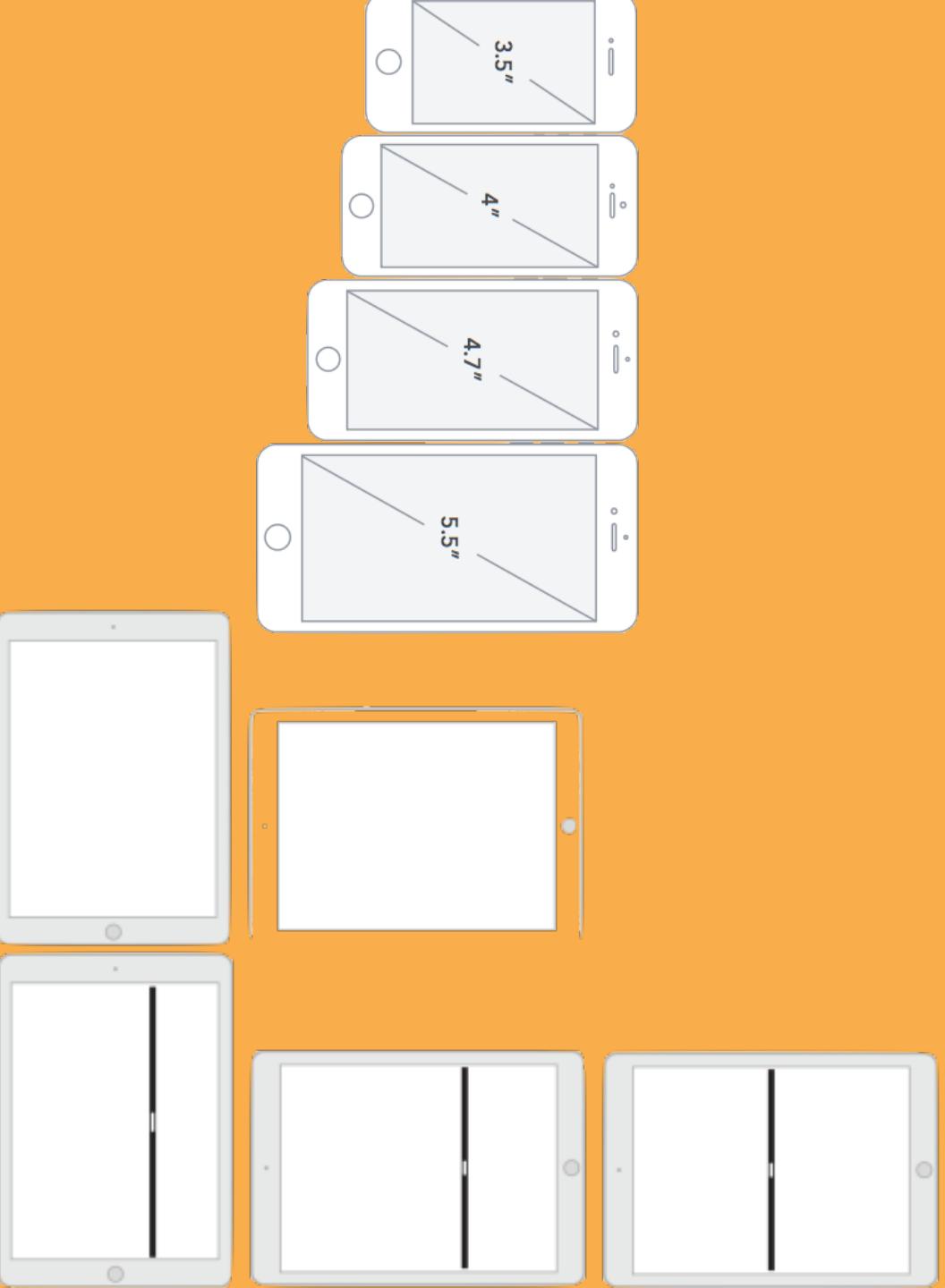
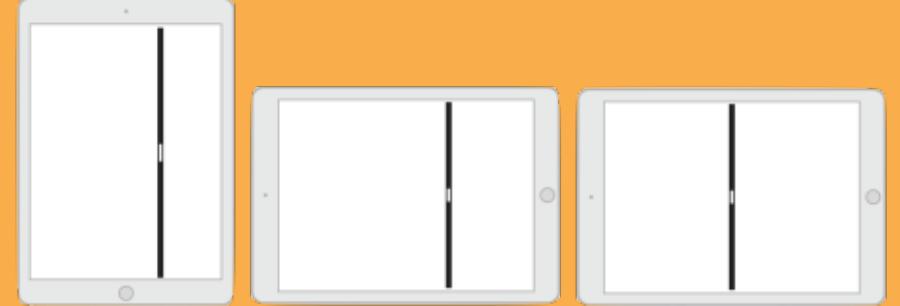
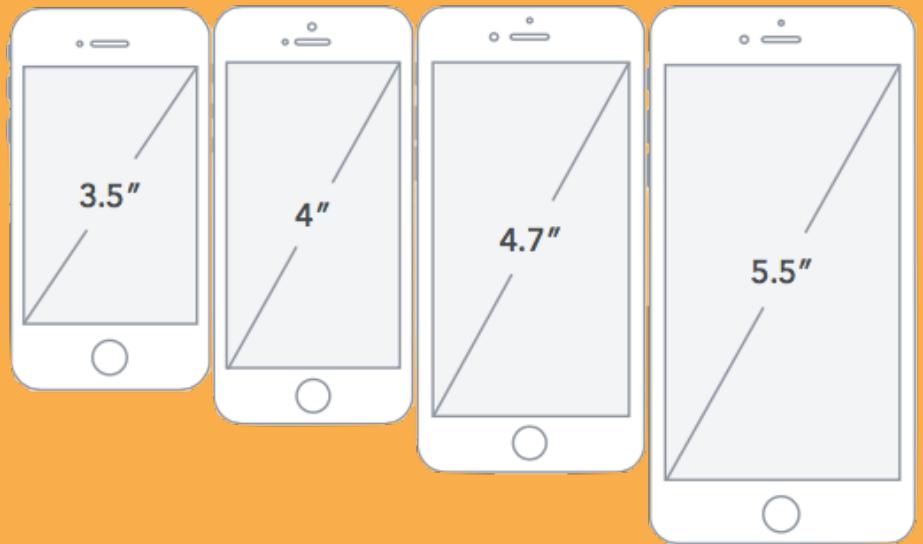
**烏來溫泉.野要獵人學野趣體驗一日遊**  
一日遊：新店捷運站→屈尺(泰雅遷移史)→泰雅博物館館內導覽→野...

**泰雅生態文化體驗(二日遊)**  
第一天：台北火車站→桃園火車站→桃園縣原住民文化會館→高義蘭...

**sample code**  
**<http://bit.ly/1WzHWMk>**

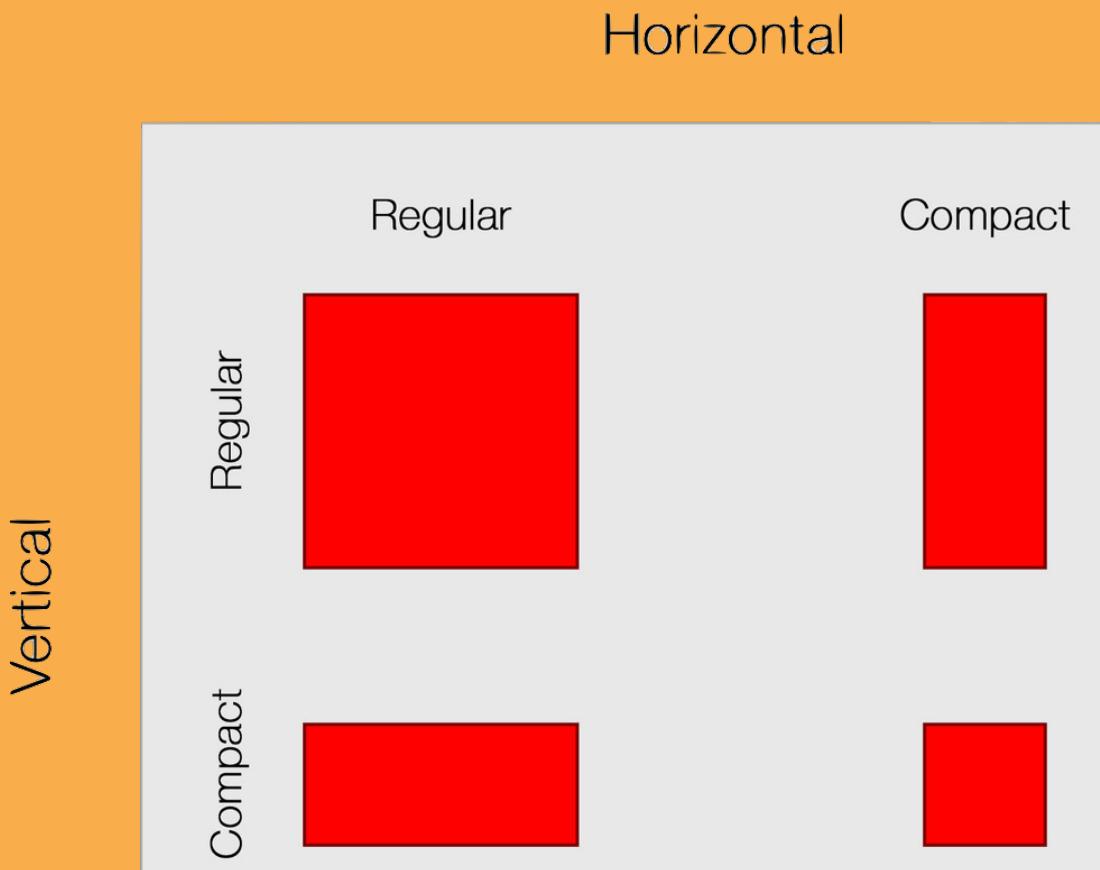
**test image url**  
**400k: <http://bit.ly/1WzIXUC>**  
**5M: <http://bit.ly/1WzJ1DT>**  
**10M: <http://bit.ly/24nDwHx>**

# SIZE CLASSES

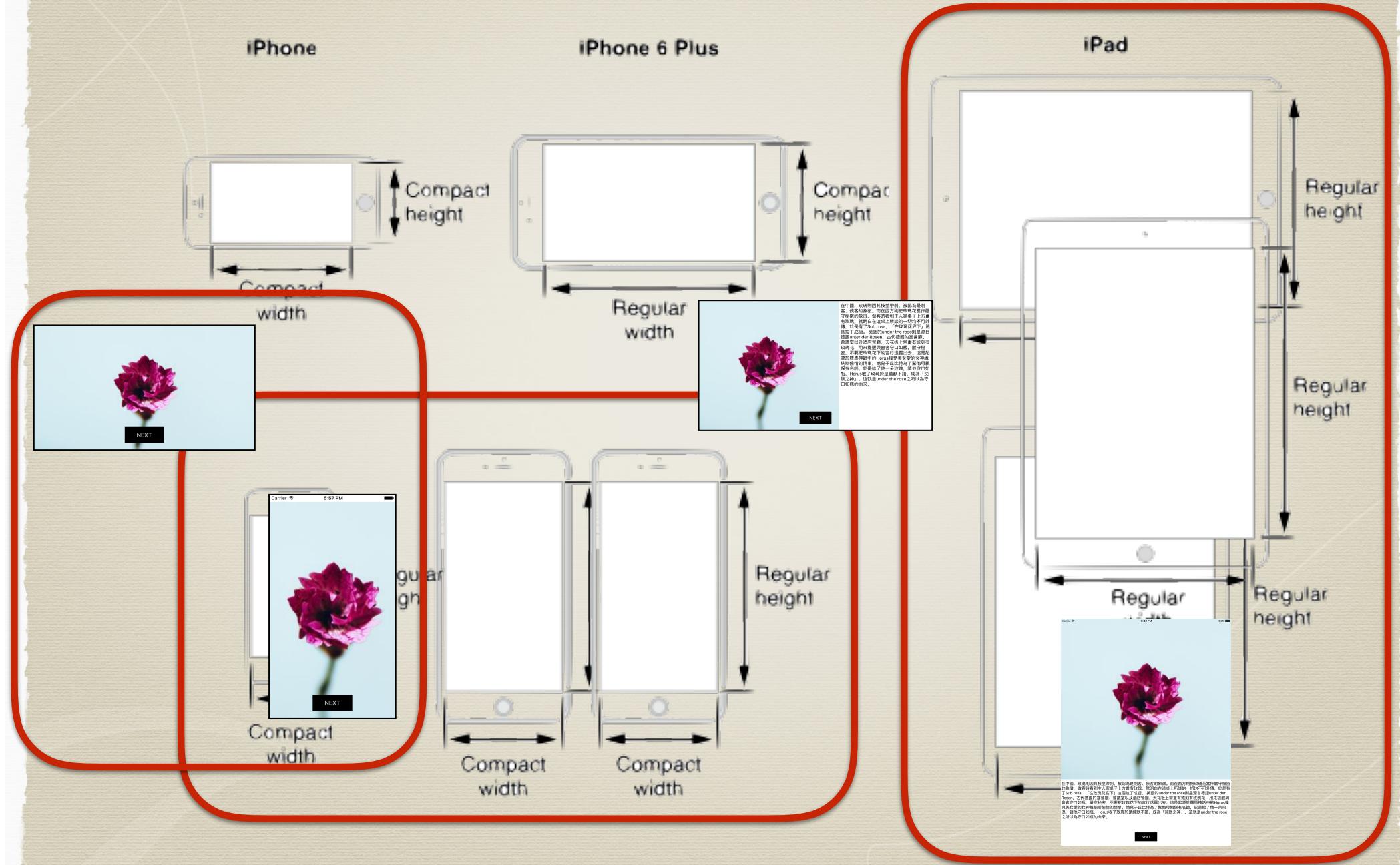


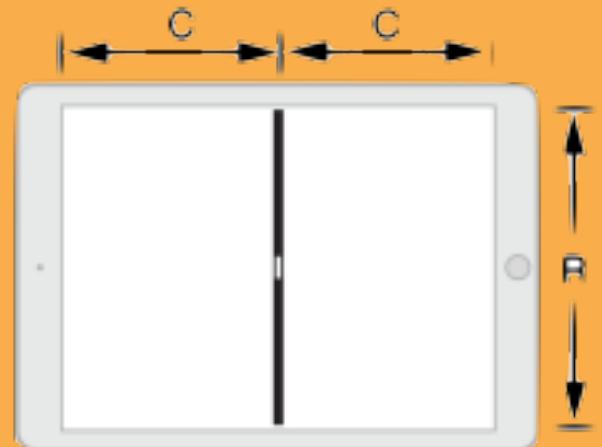
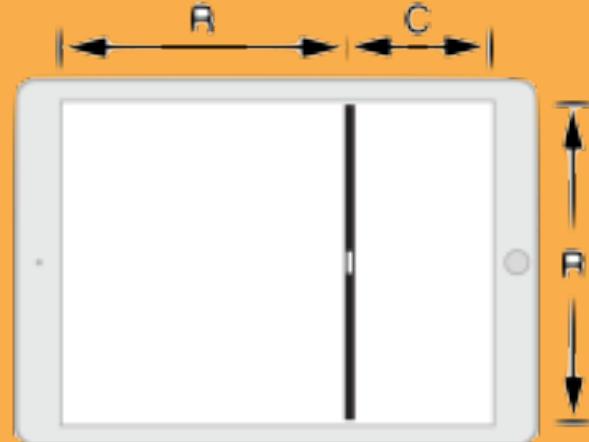
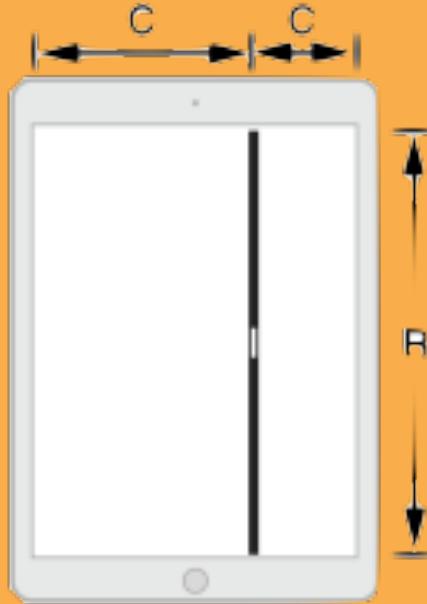
# Size Classes

- iOS8 引進的一種新概念,拋棄傳統意義上我們配置UI元件時的具體位置與尺寸,把螢幕的寬和高只分成兩種情況:Compact-緊湊, Regular-正常.



# Size Classes





# Sample



NEXT

這個拉丁成語。英語的 under the rose 則是源自德語 unter der Rosen，古代德國的宴會廳、會議室以及酒店餐廳，天花板上常畫有或刻有玫瑰花，用來提醒與會者守口如瓶，嚴守秘密，不要把玫瑰花下的言行透露出去。這是起源於羅馬神話中的 Horus 撞見美女愛的女神維納斯偷情的情事，她兒子丘比特為了幫他母親保有名節，於是給了他一朵玫瑰，請他守口如瓶，Horus 收了玫瑰於是緘默不語，成為「沈默之神」，這就是 under the rose 之所以為守口如瓶的由來。

NEXT

在中國，玫瑰則因其枝莖帶刺，被認為是刺客、俠客的象徵。而在西方則把玫瑰花當作嚴守秘密的象徵，做客時看到主人家桌子上方畫有玫瑰，就明白在這桌上所談的一切均不可外傳，於是有了 Sub rosa，「在玫瑰花底下」這個拉丁成語。英語的 under the rose 則是源自德語 unter der Rosen，古代德國的宴會廳、會議室以及酒店餐廳，天花板上常畫有或刻有玫瑰花，用來提醒與會者守口如瓶，嚴守秘密，不要把玫瑰花下的言行透露出去。這是起源於羅馬神話中的 Horus 撞見美女愛的女神維納斯偷情的情事，她兒子丘比特為了幫他母親保有名節，於是給了他一朵玫瑰，請他守口如瓶，Horus 收了玫瑰於是緘默不語，成為「沈默之神」，這就是 under the rose 之所以為守口如瓶的由來。

# Step 1 - UI

- New single view application
- Add Resource into project
- storyboard (autolayout, attribute)
  - UIImageView
  - UIButton



# Step 1 - Data, Action

- 由 plants.plist 讀入資料

```
NSBundle.mainBundle().pathForResource(  
    "plants", ofType: "plist")
```

```
NSArray(contentsOfFile: filePath)
```

- 加入 UIImageView IBOutlet

- 加入 UIButton IBAction

- 顯示下一張圖檔

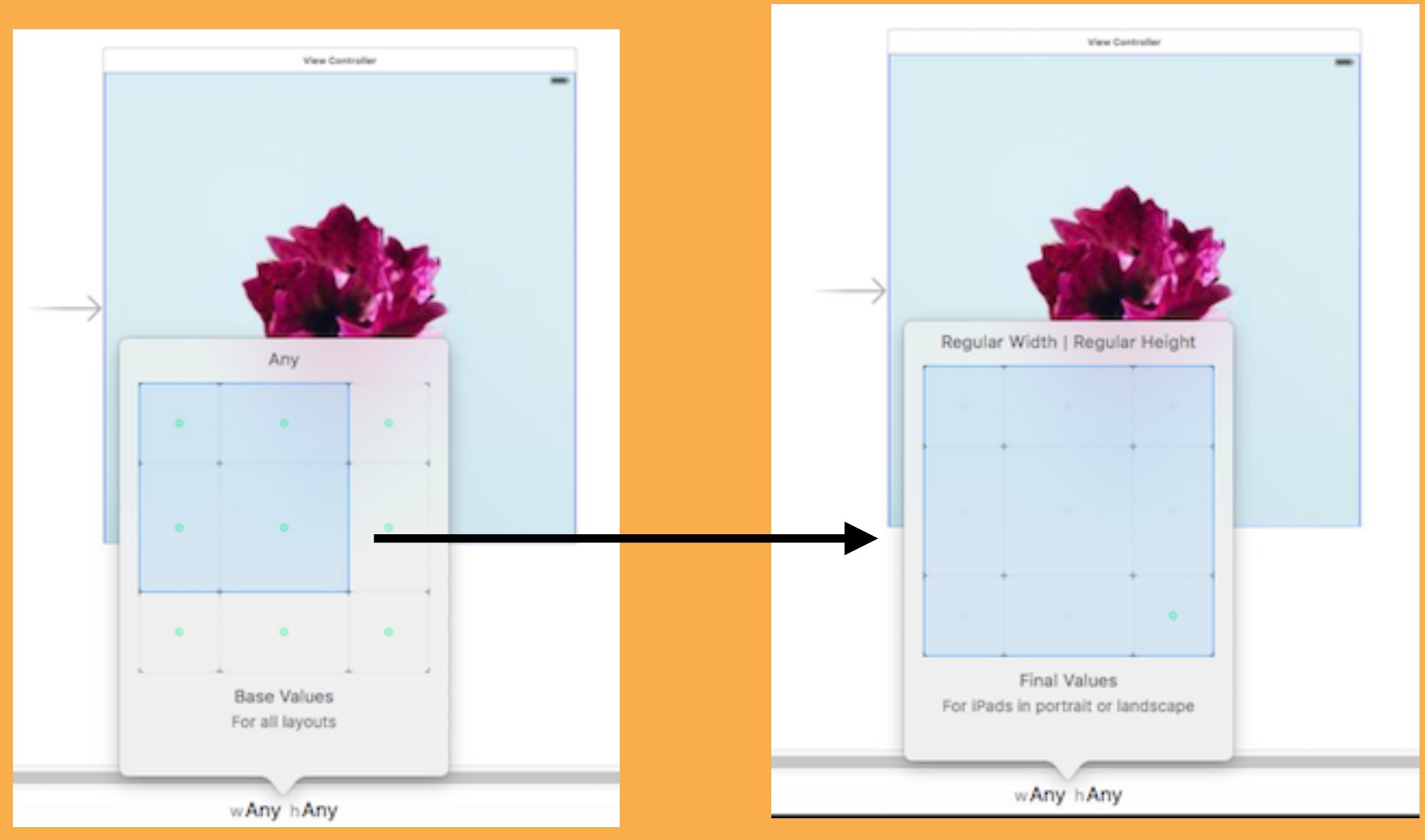


# Step 2 - UI

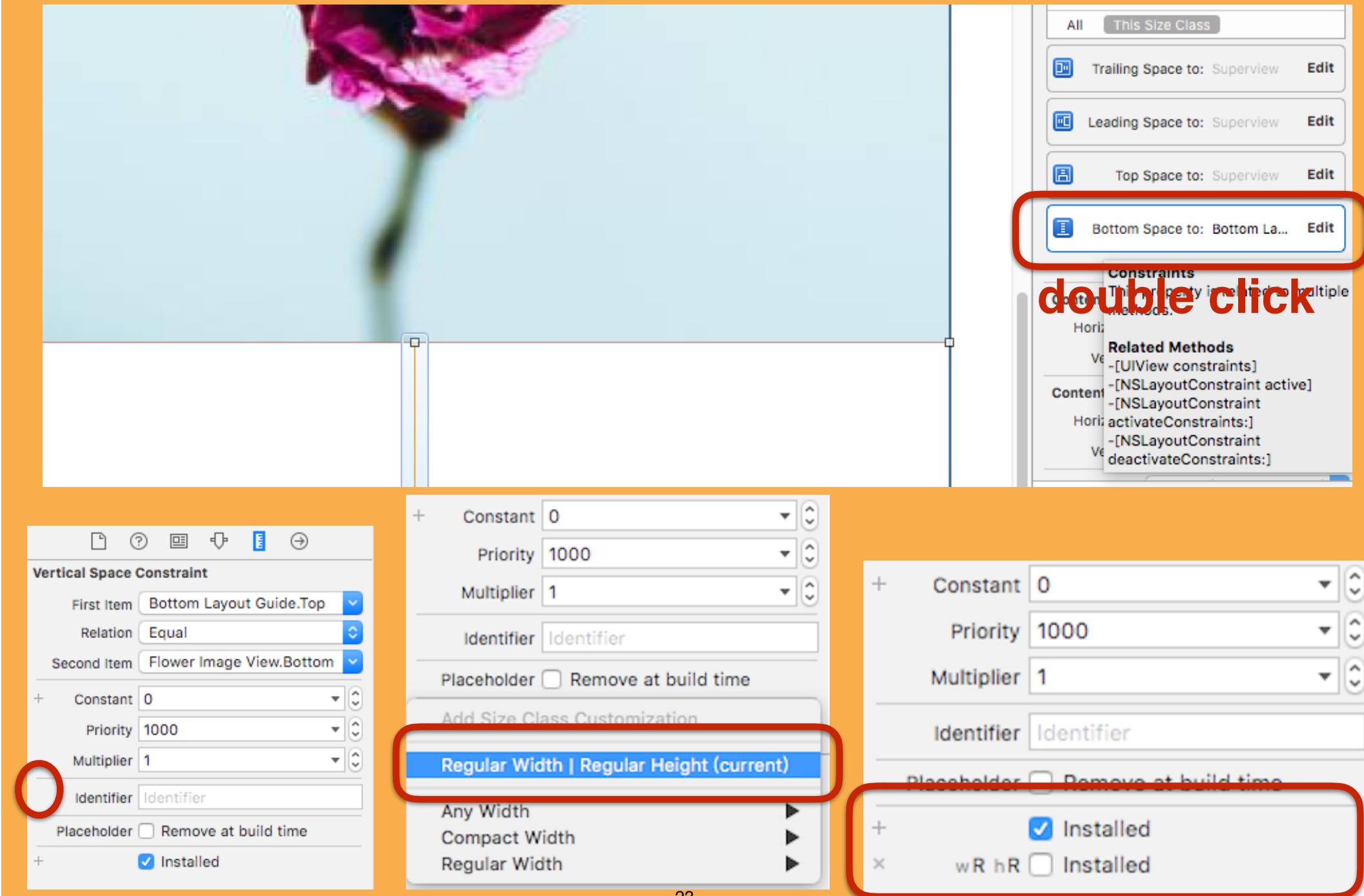
- 切換 size classes = wRegular hRegular (iPad)
- 調整 UIImageView constraints
- 加入 UITextView (and constraints)



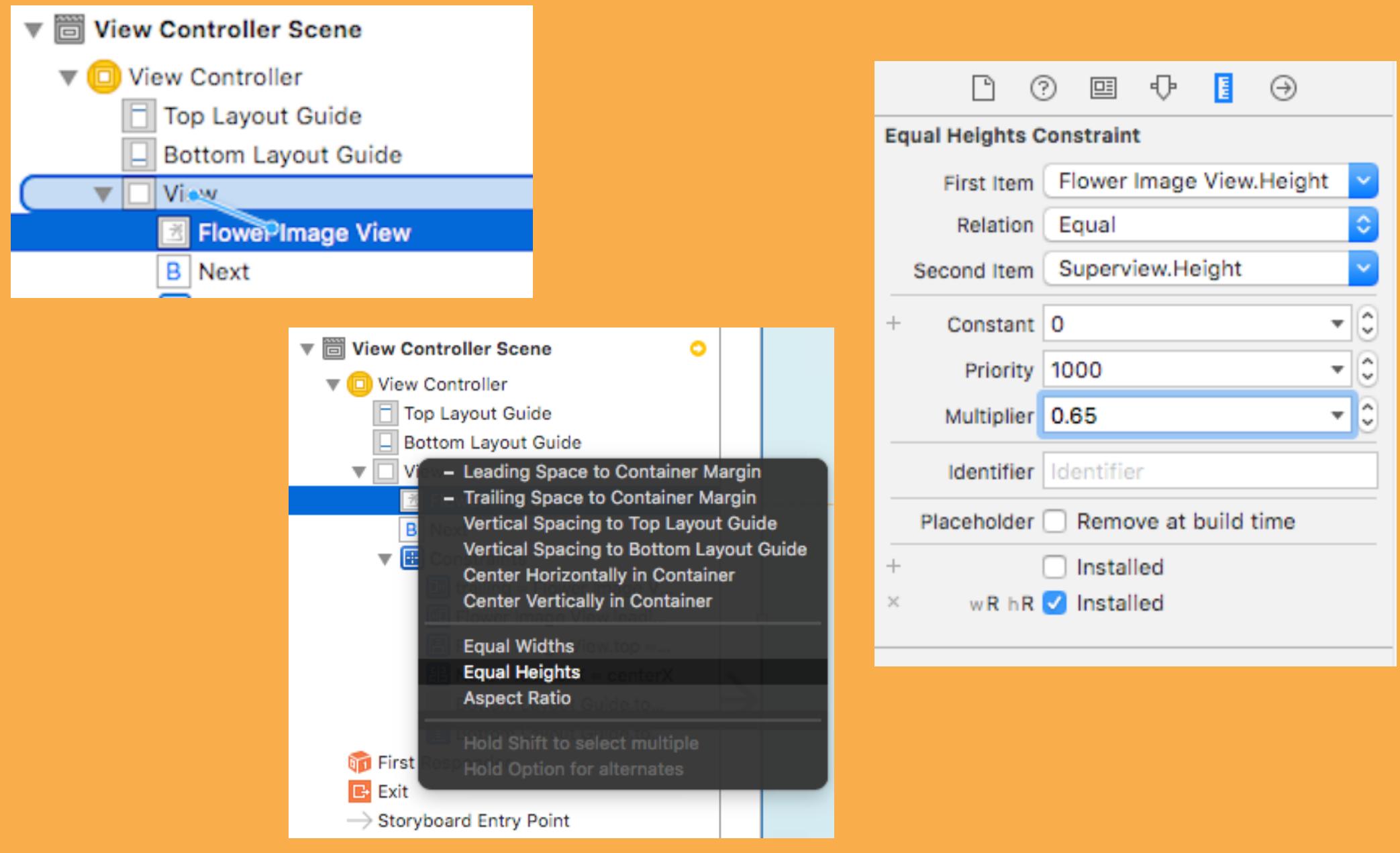
# 切換 size classes = wRegular hRegular (iPad)



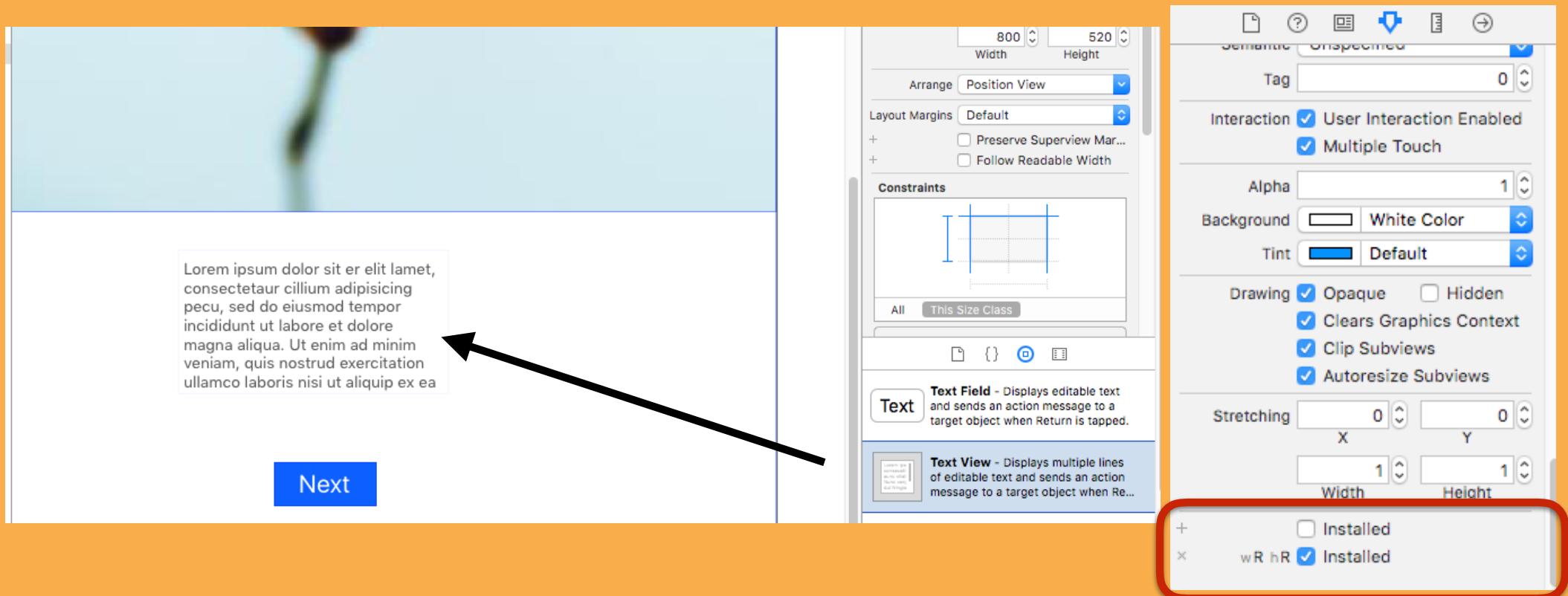
# uninstall image view bottom constraint



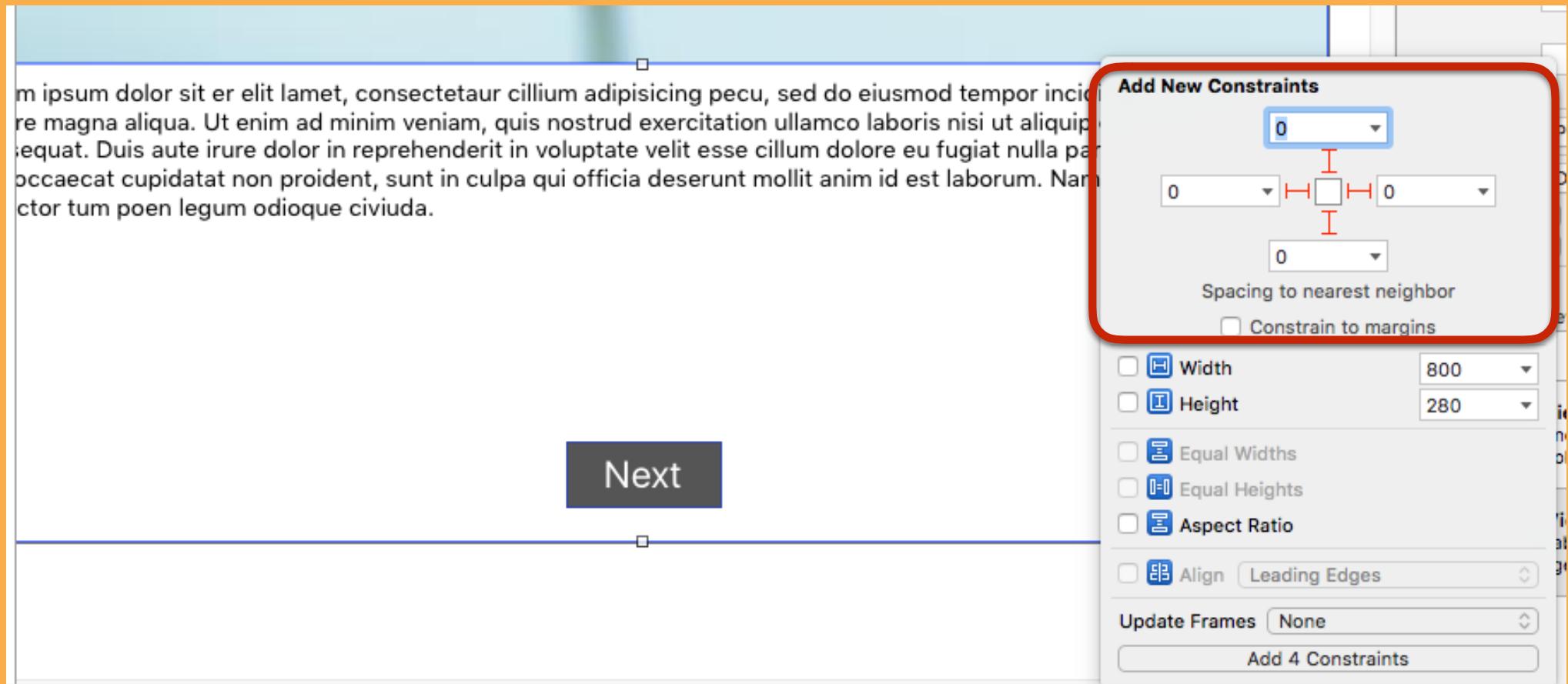
# 設定 image view 的高度是 super view 的 0.65



# 加入 UITextView



# 設定 UITextView constraints



# Step 2 - Data, Action

- 加入 UITextView IBOutlet

- 顯示圖檔 description

```
descriptionView.text = flower["description"] as! String
```

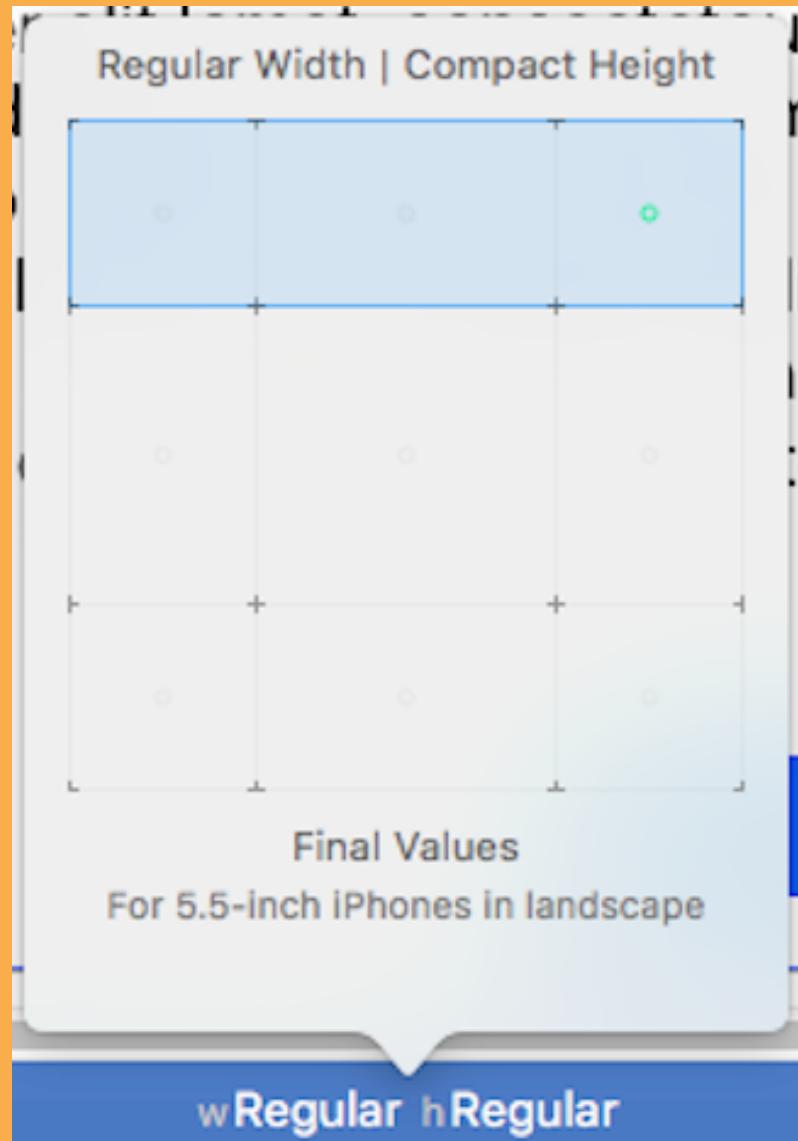


# Step 3 - UI

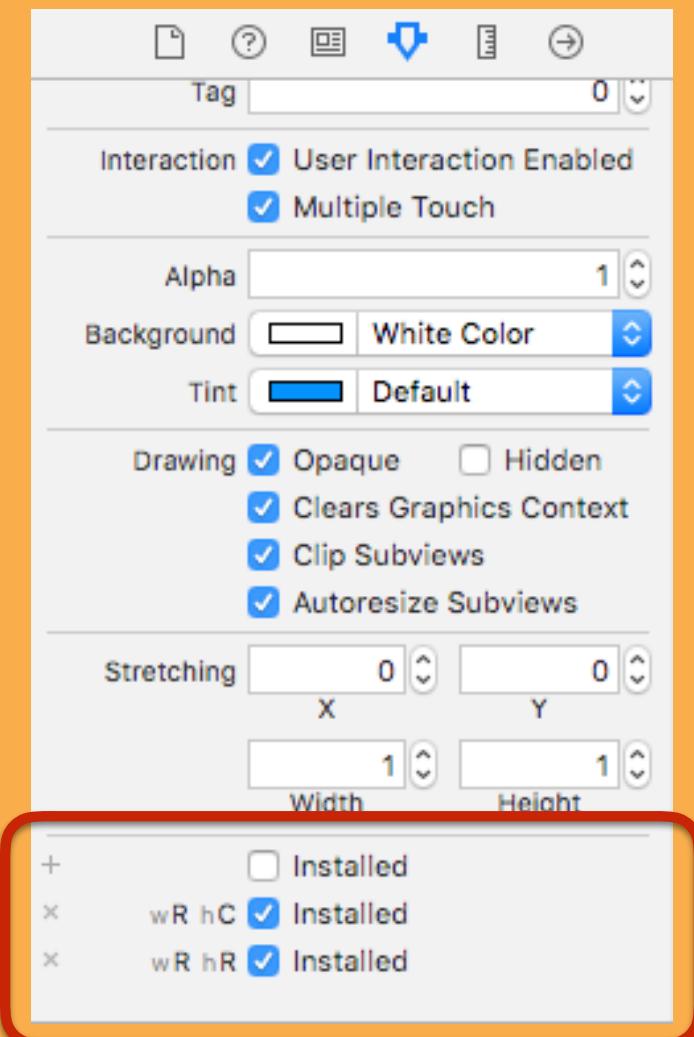
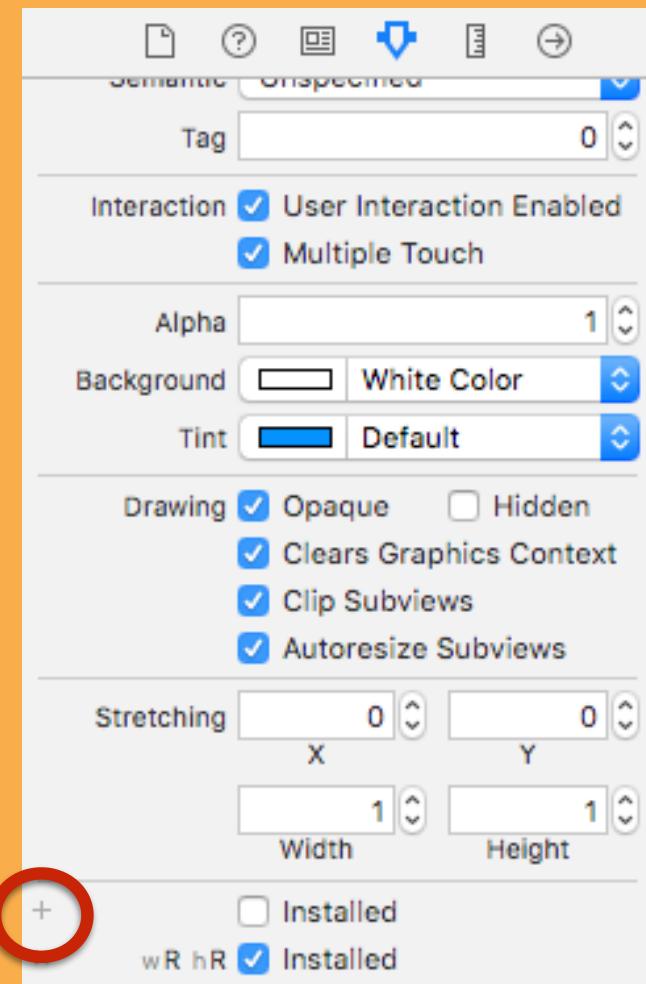
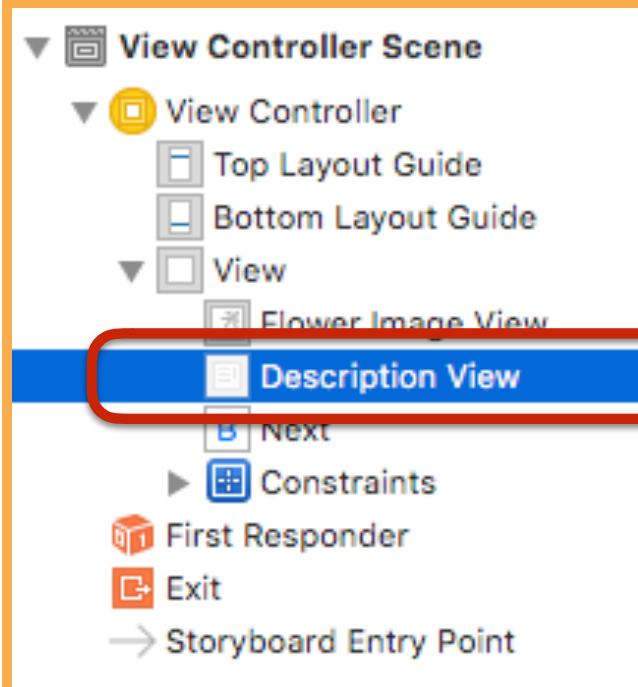
- 切換 size classes = wRegular hCompact (iPhone6+)
- Add UITextView (and constraints)
- 調整 UIImageView constraints



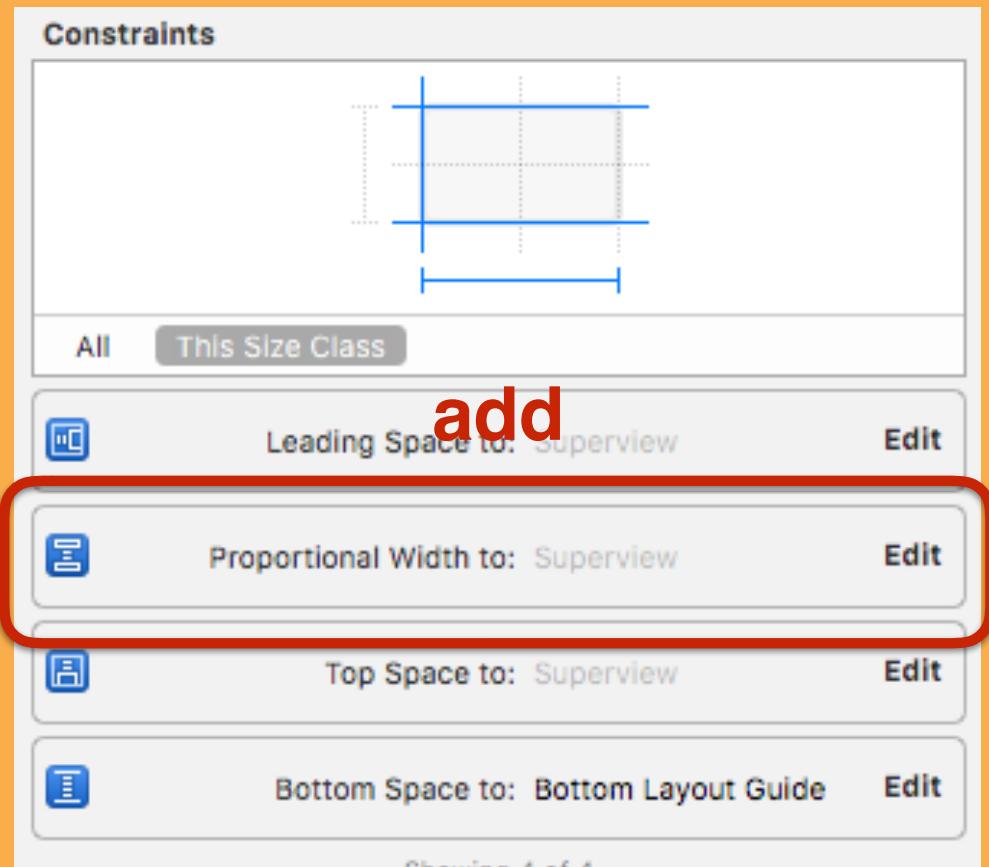
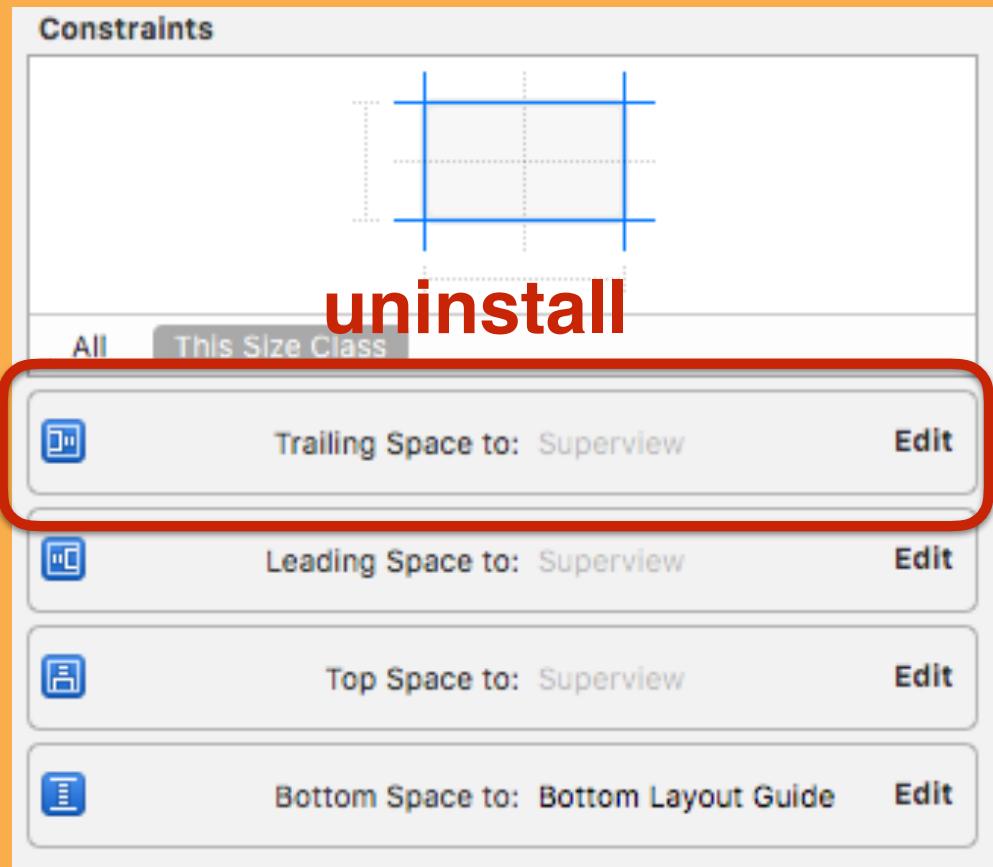
切換 size classes = wRegular hCompact (iPhone6+)



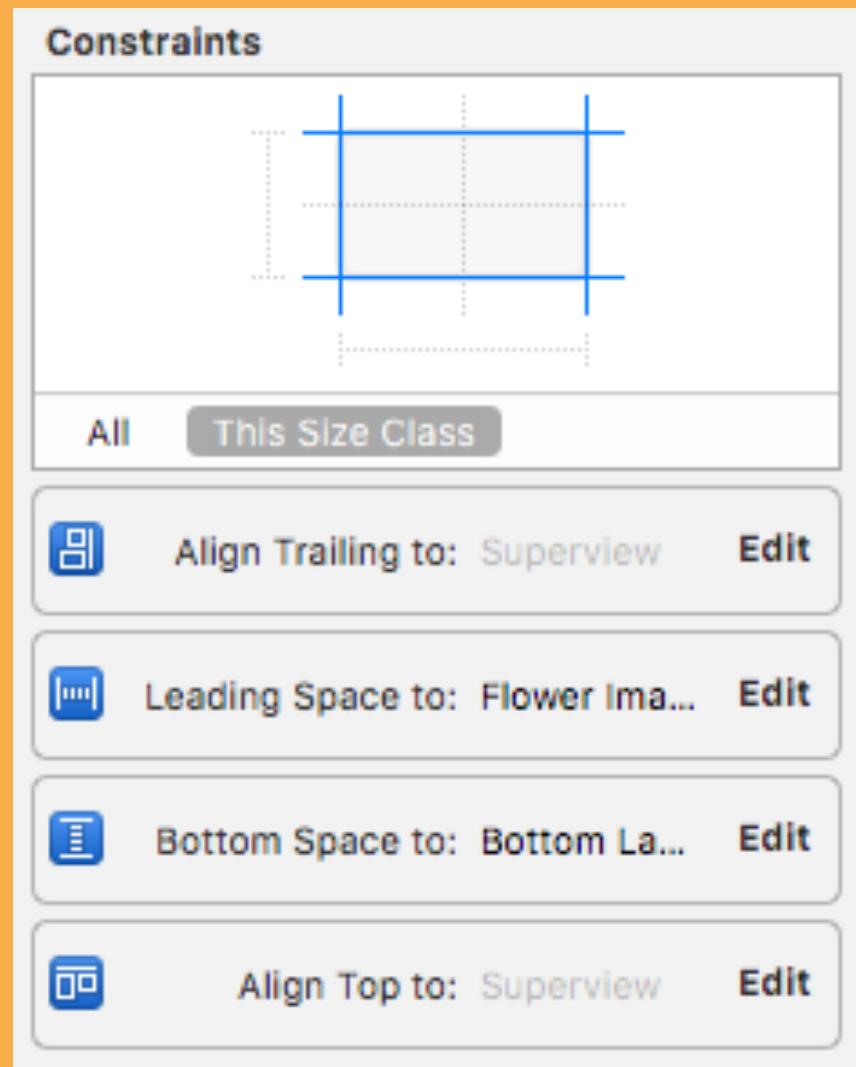
# Add UITextView



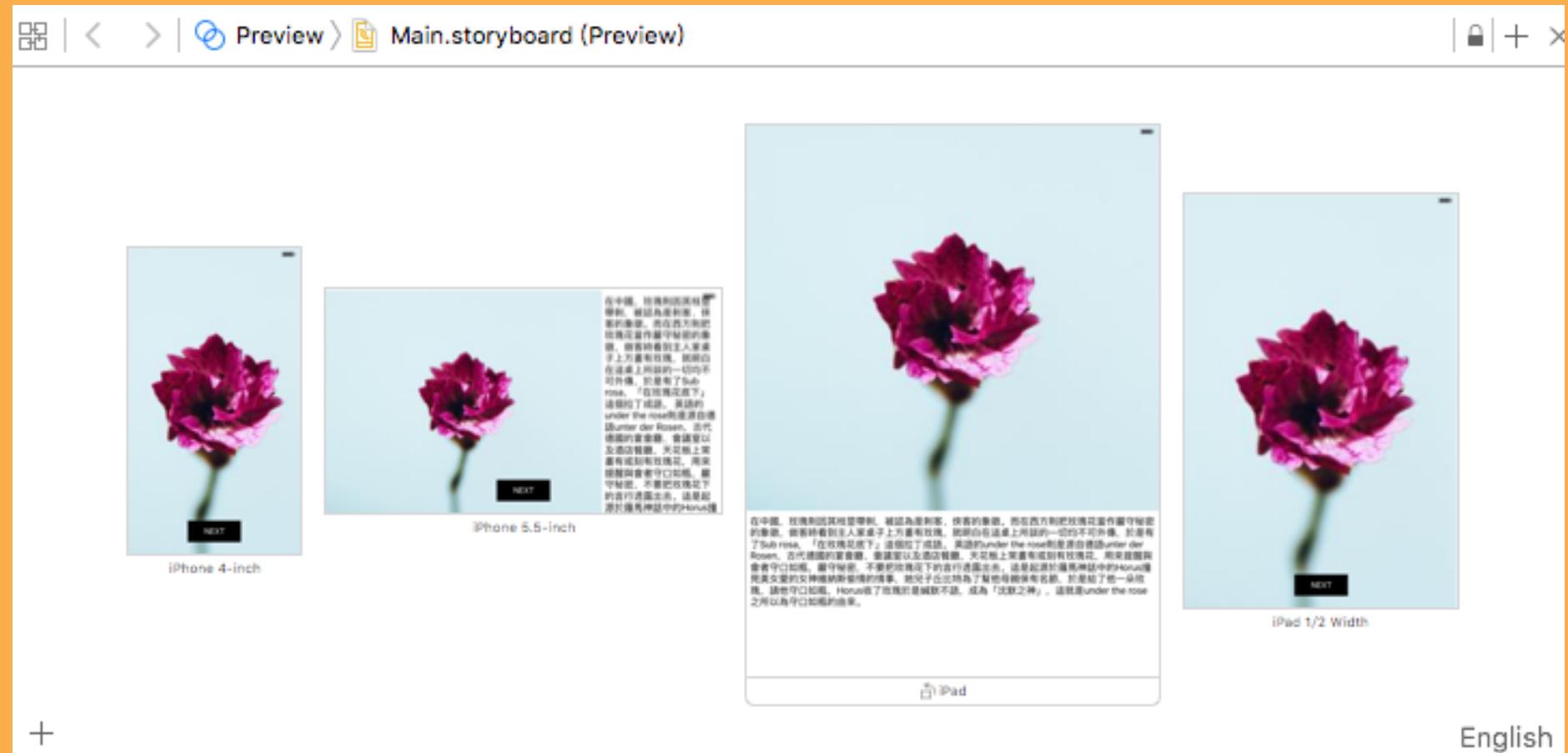
# 調整 UIImageView constraints

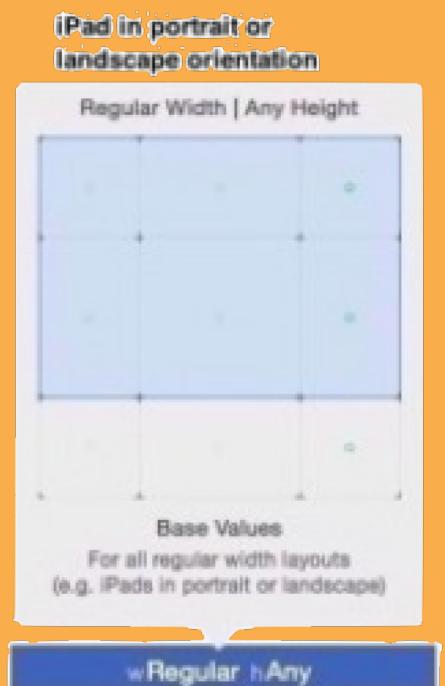
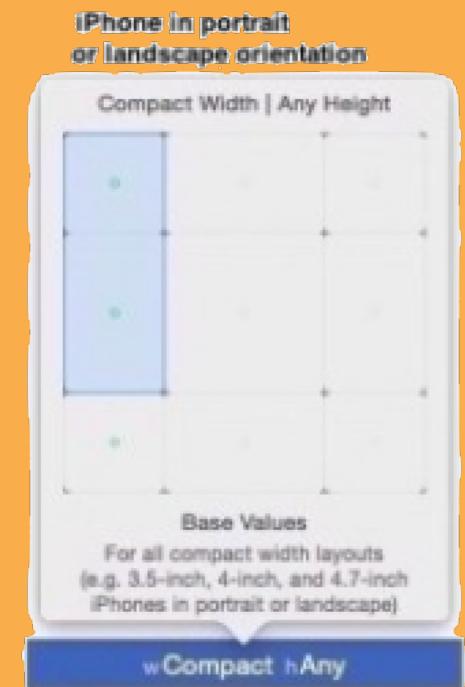
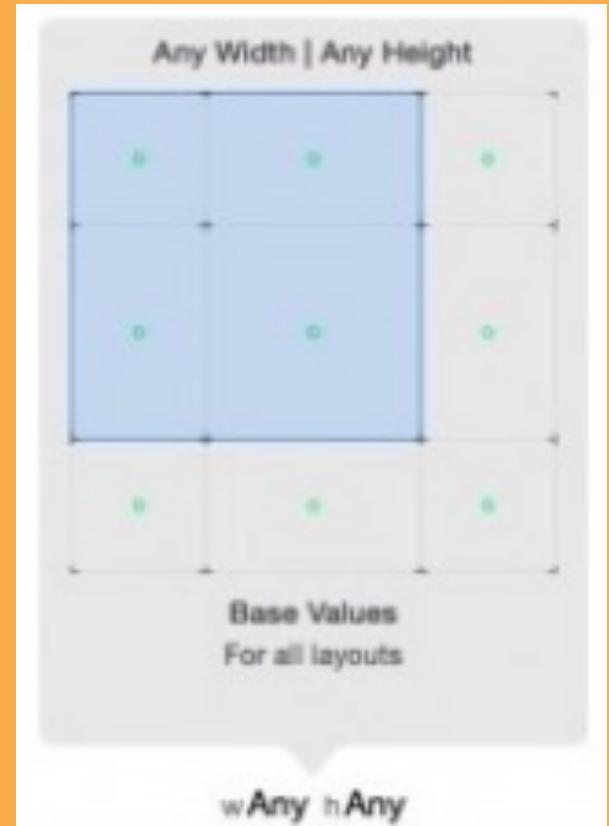


# 調整 UITextView constraints

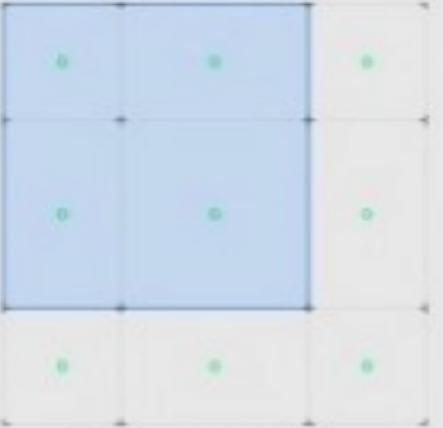


# Preview in interface builder





### Any Width | Any Height



Base Values  
For all layouts

w Any h Any

### Regular Width | Regular Height



Final Values  
For iPads in portrait or landscape

w Regular h Regular

### Regular Width | Compact Height

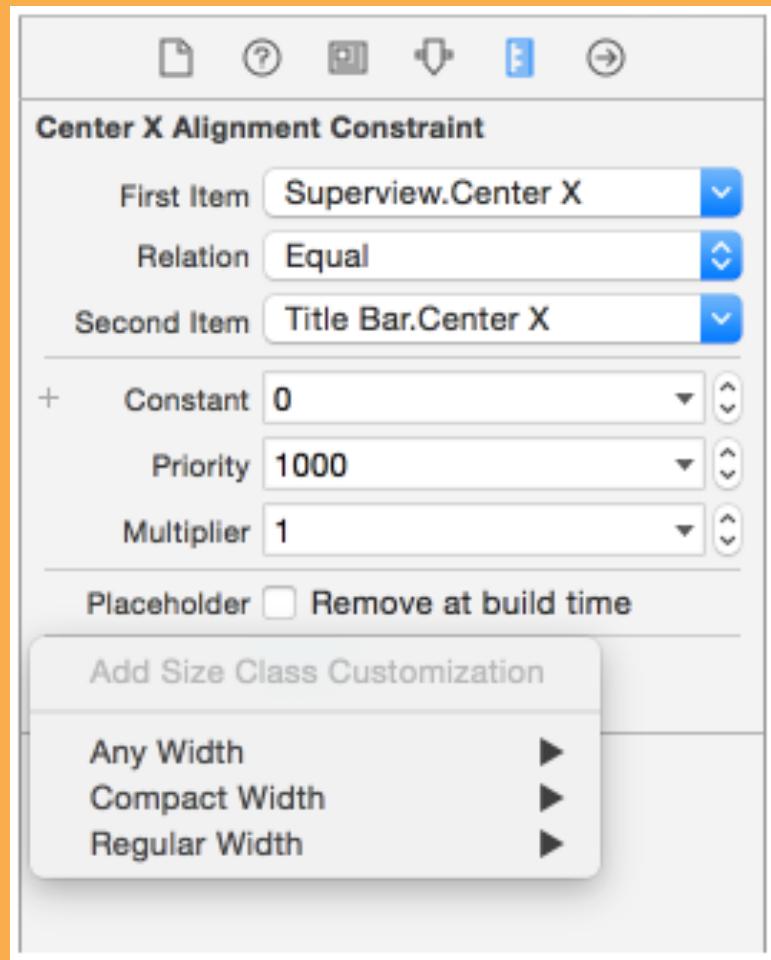


Final Values  
For 5.5-inch iPhones in landscape

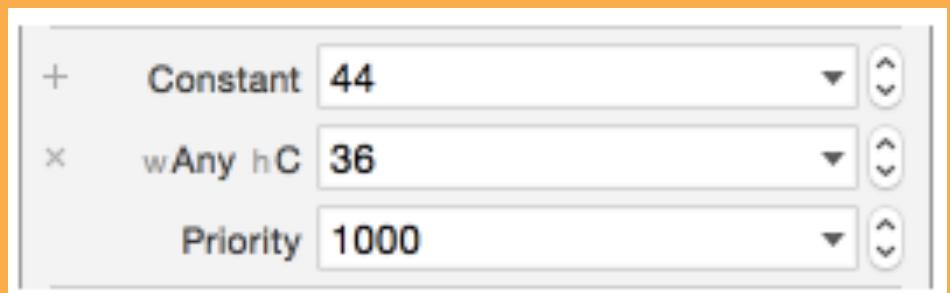
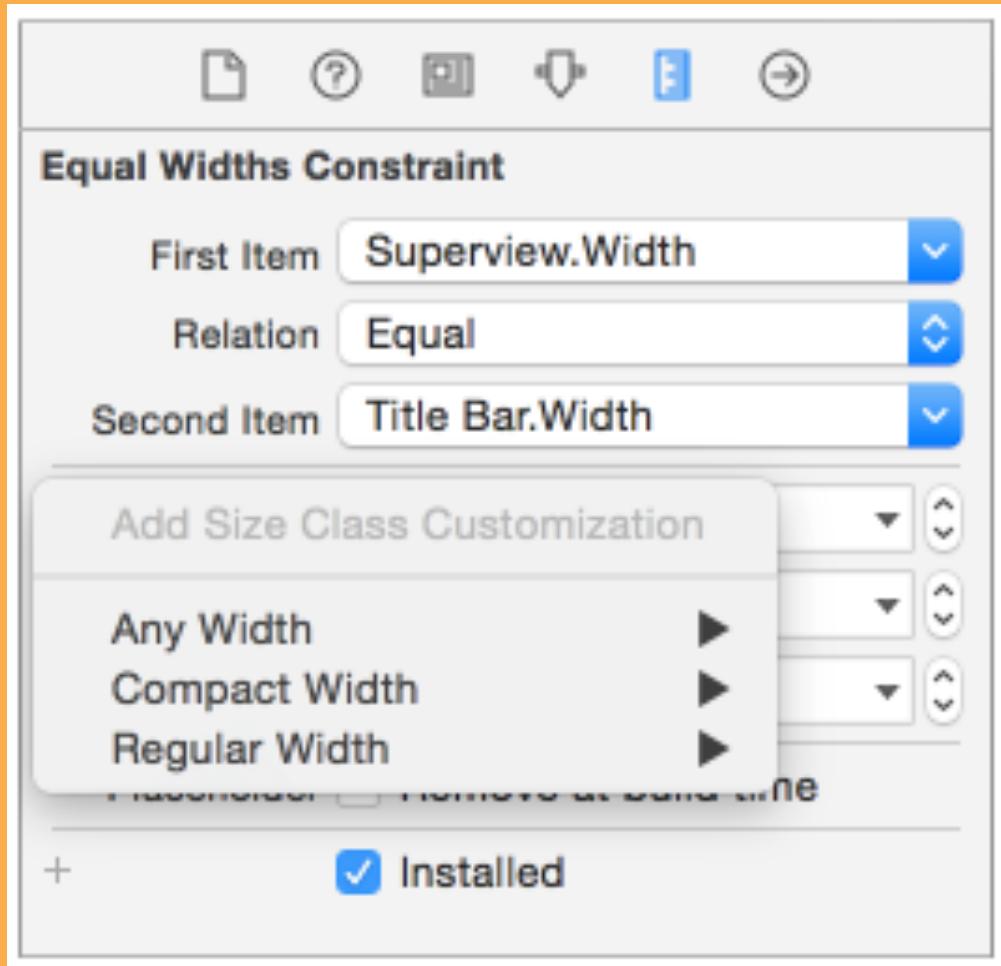
w Regular h Compact



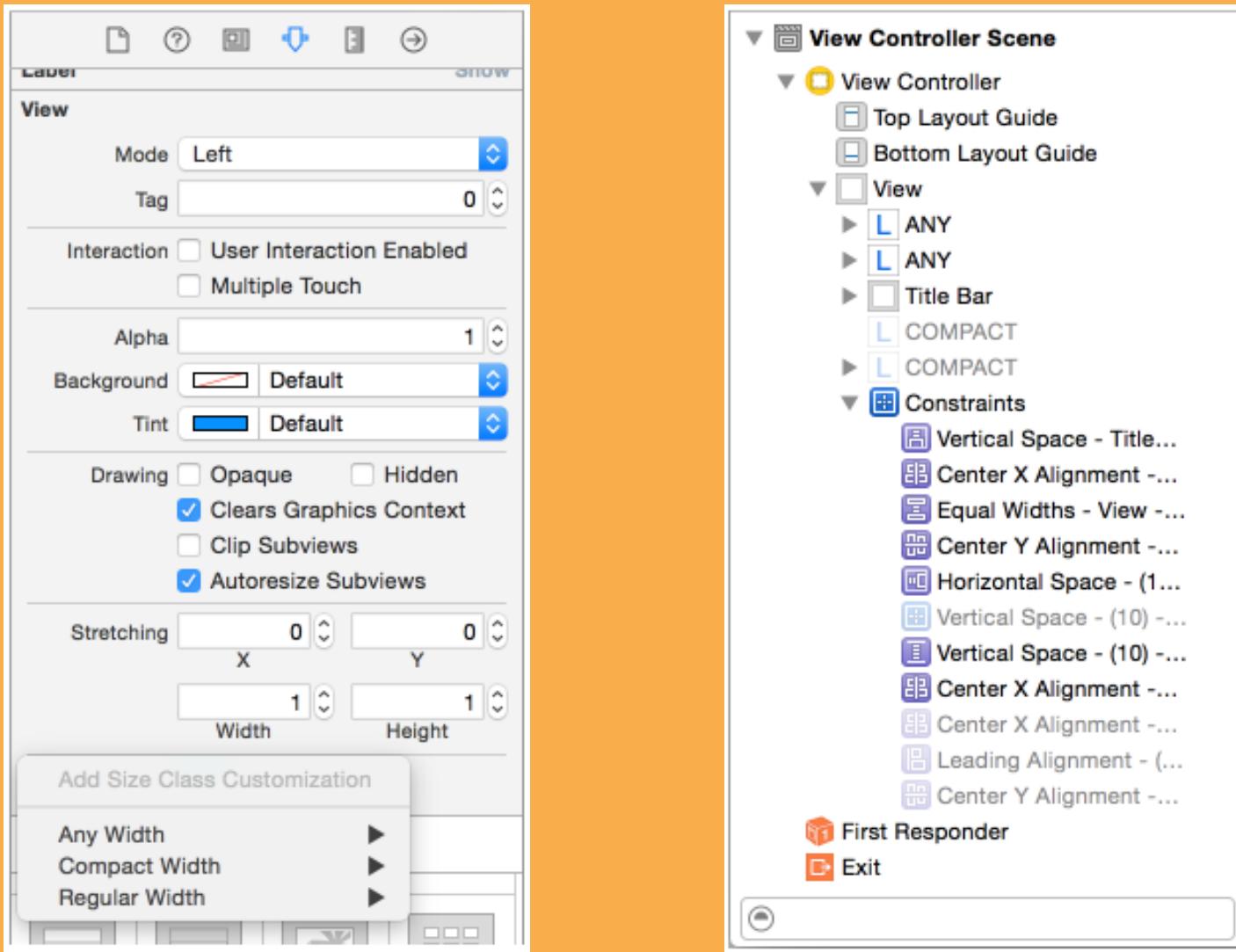
# Installing and Uninstalling Constraints for a Size Class



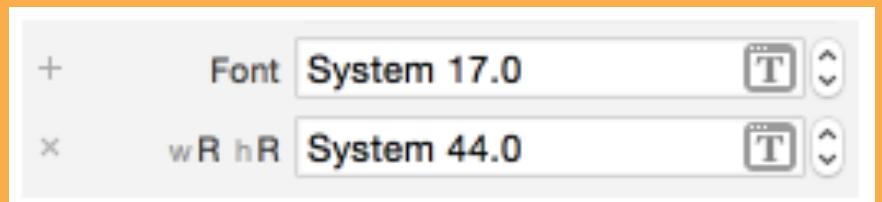
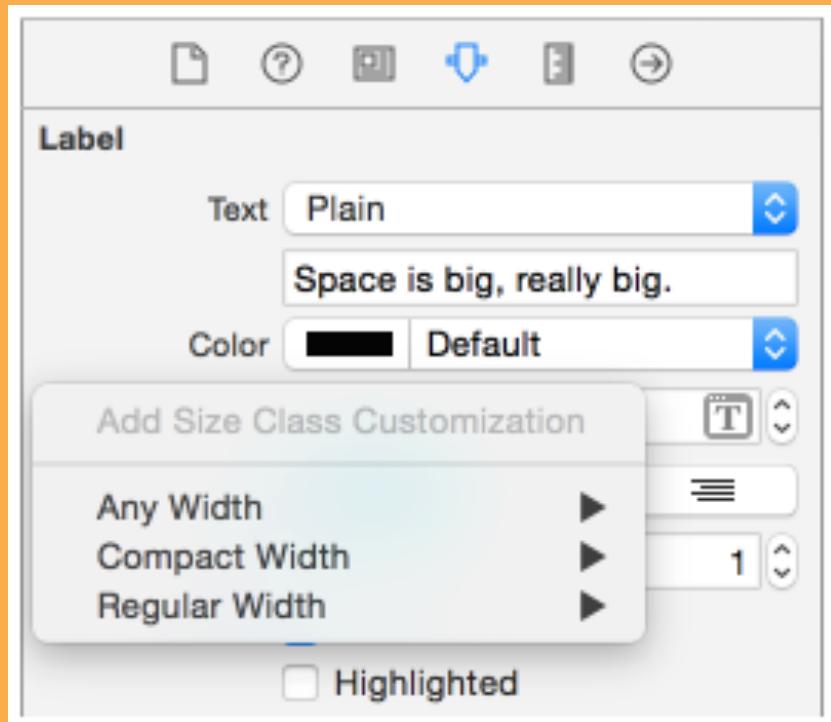
# Changing Constraint Constants for a Size Class



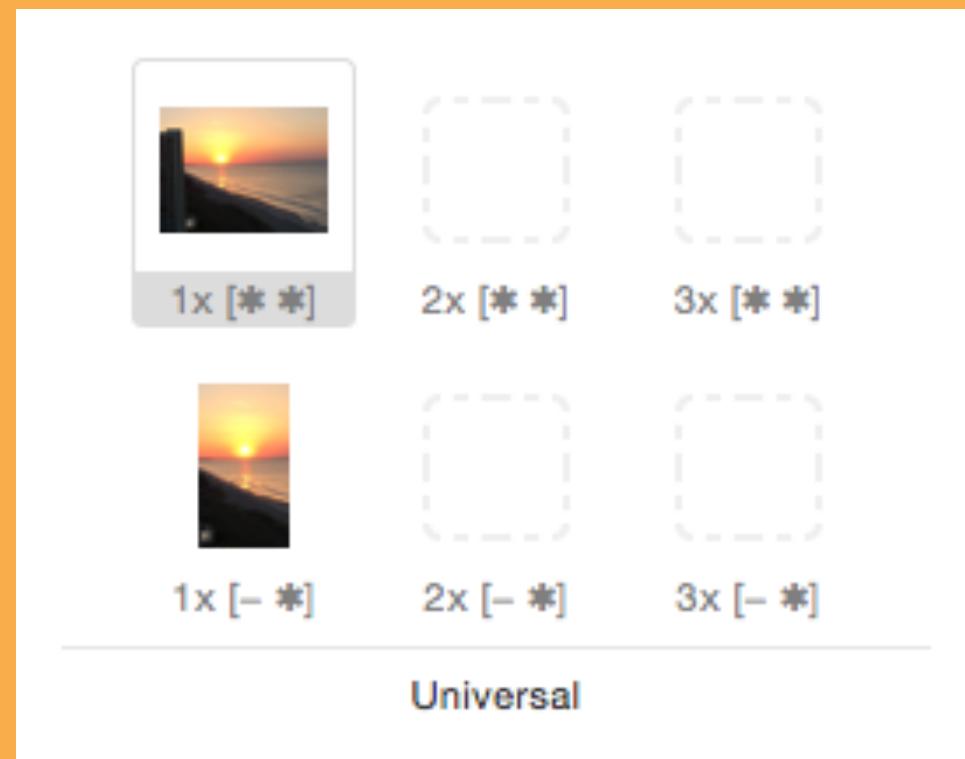
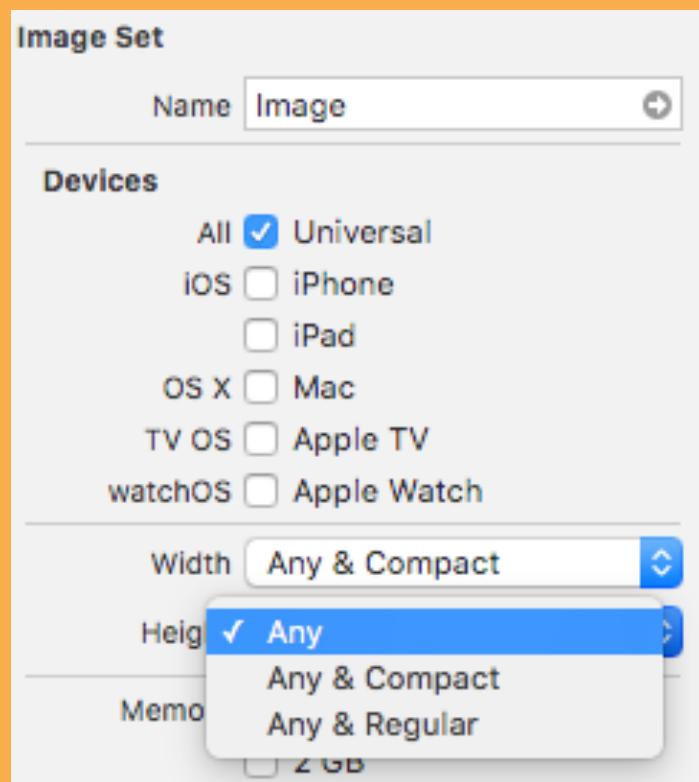
# Installing and Uninstalling Views for a Size Class



# Changing the Font for a Size Class

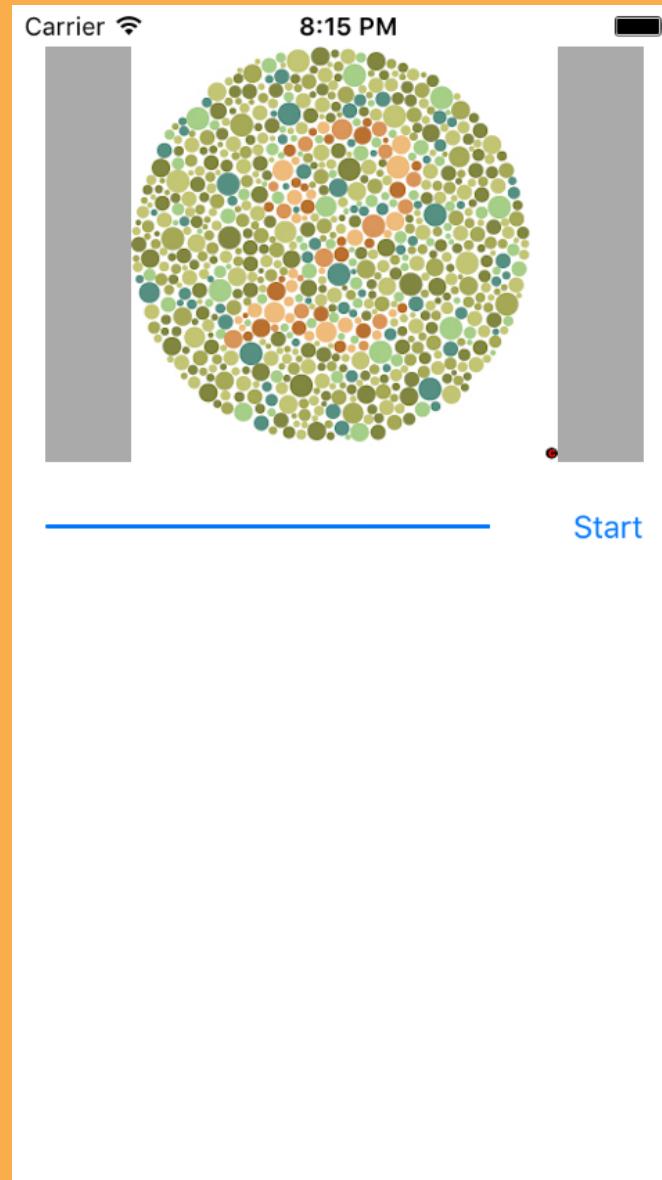
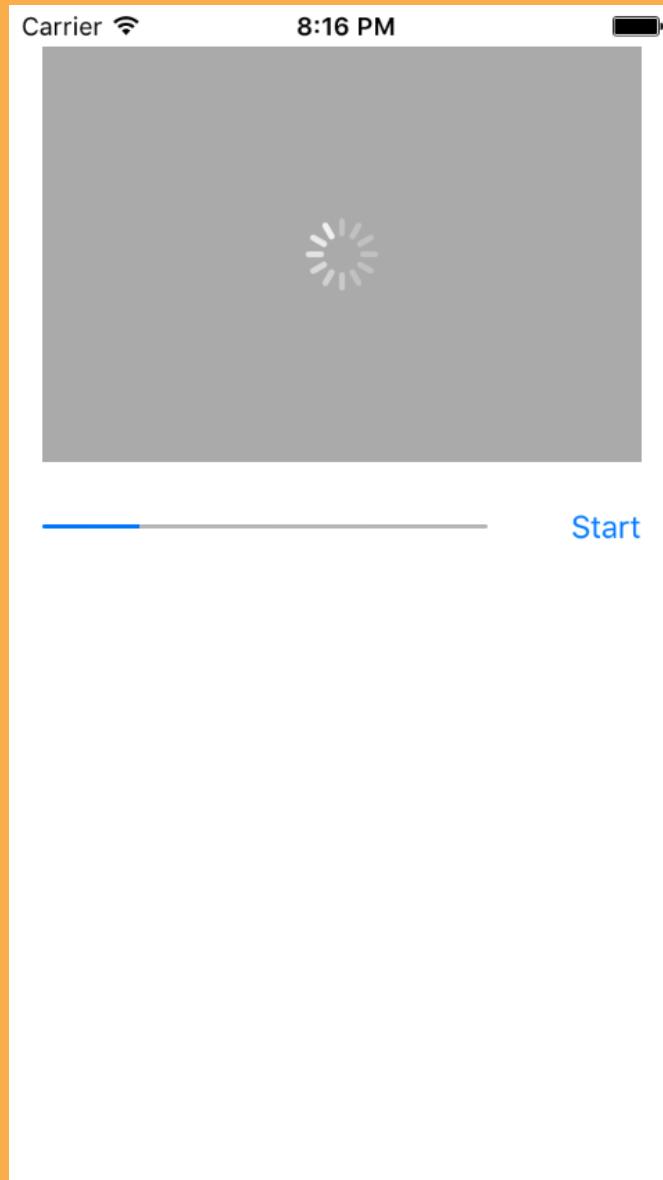


# UIImage Asset



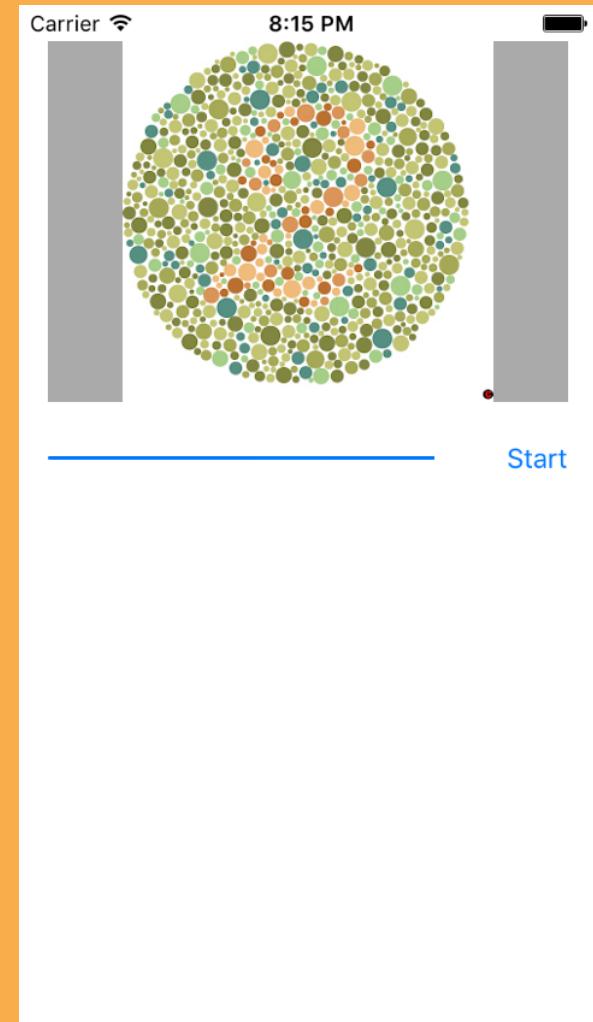
# NSURLSession

# Download image sample



# UI

- New single view application
- UIImageView
- UIProgressView
- UIButton
- UIActivityIndicatorView
- constraints and IBOutlets
- Add ATS



# ATS

Info.plist	M	Bundle OS Type code	String	APPL
Products		Bundle versions string, short	String	1.0
		Bundle creator OS Type code	String	????
		Bundle version	String	1
		Application requires iPhone environment	Boolean	YES
		Launch screen interface file base name	String	LaunchSc
		Main storyboard file base name	String	Main
		► Required device capabilities	Array	(1 item)
		► Supported interface orientations	Array	(3 items)
		► Supported interface orientations (iPad)	Array	(4 items)
		▼ App Transport Security Settings	Dictionary	(1 item)
		Allow Arbitrary Loads	Boolean	YES

```
defaultSessionConfiguration
```

```
.timeoutIntervalForRequest = 15
```

```
.HTTPAdditionalHeaders = {"Accept": "application/json"}
```

```
.....
```

```
NSURLRequest
```

```
https://dl.dropboxusercontent.com/u/23305422/testImage-10M.png
```

Configuration  
(NSURLSessionConfiguration)

NSURLSession

Delegate

Queue  
Task  
Task  
Task

```
NSURLSessionDownloadTask
```

```
func URLSession(session: NSURLSession, downloadTask: NSURLSessionDownloadTask,  
didWriteData bytesWritten: Int64, totalBytesWritten: Int64,  
totalBytesExpectedToWrite: Int64)
```

```
func URLSession(session: NSURLSession, downloadTask: NSURLSessionDownloadTask,  
didFinishDownloadingToURL location: NSURL)
```

# configurations

```
configuration.timeoutIntervalForRequest = 15  
  
configuration.HTTPAdditionalHeaders =  
["Accept":"application/json"];  
  
configuration.HTTPAdditionalHeaders =  
["Authorization":authorizationHeader]  
  
configuration.HTTPMaximumConnectionsPerHost = 1  
  
configuration.requestCachePolicy =  
NSURLRequestUseProtocolCachePolicy;
```

# URL Request

- \* **Simple requests:** use theNSURLSession API to retrieve the contents from an NSURL object directly.

```
let url = NSURL(string: "http://data.taipei/...")  
let task = session.dataTaskWithURL(url!) { (data , response, error ) -> Void in  
    ....  
}
```

- \* **Complex requests:** provide an NSURLRequest object to NSURLSession.

```
let url = NSURL(string: "http://data.taipei/...")  
let request = NSMutableURLRequest(URL: requestUrl!)  
request.HTTPMethod = "POST"  
....  
let task = session.dataTaskWithRequest(request) { (data, response, error ) -> Void  
in  
....}
```

# NSURLSessionDownload Task

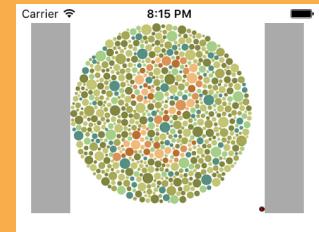
- \* retrieve data in the form of a **file**. (directly write the server's response data to a temporary file)
- \* support background downloads while the app is not running
- \* support pause (cancel) download tasks and resume them later

```
func URLSession(_ session: NSURLSession, downloadTask downloadTask:  
NSURLSessionDownloadTask, didWriteData bytesWritten: Int64, totalBytesWritten  
totalBytesWritten: Int64, totalBytesExpectedToWrite totalBytesExpectedToWrite:  
Int64)
```

```
func URLSession(_ session: NSURLSession, downloadTask downloadTask:  
NSURLSessionDownloadTask, didFinishDownloadingToURL location: NSURL)
```

# start button IBAction

```
let sessionConfiguration =  
    NSURLSessionConfiguration.defaultSessionConfiguration();  
  
session = NSURLSession(configuration: sessionConfiguration,  
                      delegate: self, delegateQueue: nil)  
  
let downloadTask = session.downloadTaskWithURL(NSURL(string:  
    "https://dl.dropboxusercontent.com/u/23305422/  
    testImage-10M.png")!)  
  
progressbar.progress = 0.0  
busyIndicator.startAnimating()  
imageView.image=nil  
startBtn.enabled = false  
  
downloadTask.resume()
```



# NSURLSessionDownloadDelegate

```
func URLSession(session: NSURLSession, downloadTask:  
NSURLSessionDownloadTask, didWriteData bytesWritten: Int64,  
totalBytesWritten: Int64, totalBytesExpectedToWrite: Int64){  
  
    dispatch_async(dispatch_get_main_queue(), { () -> Void in  
  
        self.progressbar.progress = Float(totalBytesWritten) /  
            Float(totalBytesExpectedToWrite)  
  
    })  
}
```

# NSURLSessionDownloadDelegate

```
func URLSession(session: NSURLSession, downloadTask:  
NSURLSessionDownloadTask, didFinishDownloadingToURL location:  
NSURL){  
  
    let data = NSData(contentsOfURL: location)  
    if let imageData = data {  
        dispatch_async(dispatch_get_main_queue(), { () -> Void in  
            self.imageView.image = UIImage(data: imageData)  
        })  
    }  
  
    dispatch_async(dispatch_get_main_queue(), { () -> Void in  
        self.busyIndicator.stopAnimating()  
        self.startBtn.enabled = true  
    })  
  
    session.finishTasksAndInvalidate()  
}
```

# NSURLSessionDownloadDelegate

```
func URLSession(session: NSURLSession, task: NSURLSessionTask,  
didCompleteWithError error: NSError?) {  
  
    if error != nil {  
        print("Error download data, error=\(error)")  
        session.finishTasksAndInvalidate()  
  
        dispatch_async(dispatch_get_main_queue(), { () -> Void  
            in  
                self.busyIndicator.stopAnimating()  
                self.startBtn.enabled = true  
            })  
    }  
}
```

# Data request sample



登入平臺

會員服務

平臺關鍵字

查詢

資料集下載

互動專區

活化應用

諮詢小組

M2M專區

授權條款

關於平臺

相關連結

## Open Data 顯微鏡

讓政府開放資料帶你細看醫療問題!



<http://data.gov.tw>

# 原鄉旅遊路線

資料集評分:   
平均: 4 (2 投票)

[訂閱](#) [訂閱說明](#)

資料集描述	原住民族地區之旅遊規劃路線之參考。
主要欄位說明	序號、資料日期、路線名稱、民族、所在地、交通方式、行程內容、旅遊季節、旅遊天數、行程特色、部落資訊(負責機構)、重要祭典、問候語、成行人數、每人費用、住宿地點、組合。
資料資源	<a href="#"> XML</a> <a href="#"> 檢視資料</a> <a href="#"> JSON</a> <a href="#"> 檢視資料</a> <a href="#"> CSV</a> <a href="#"> 檢視資料</a>
資料集類型	原始資料
資料集提供機關名稱	原住民族委員會

<http://data.gov.tw/node/7128>

```
{  
    "Seq": 1,  
    "DateListed": "2013/09/01",  
    "Name": "老地方展新魅力—台東部落逍遙遊",  
    "Ethnic": "魯凱族、排灣族",  
    "Region": "臺東縣金峰鄉、太麻里鄉、卑南鄉、臺東市",  
    "Access": "",  
    "TravelItinerary": "一日遊：屏東縣→達仁鄉→大武鄉→多良向陽  
    "TouristSeason": "",  
    "TravelDays": "一個太陽",  
    "TravelCharacteristics": "多良火車站 / 多良向陽薪傳木工坊",  
    "TribeInformation": "",  
    "ImportantRite": "",  
    "Greetings": "",  
    "AssociationPopulation": "",  
    "CostPerPerson": "",  
    "Accommodation": "",  
    "Group": "2013年60條原住民族深度旅遊路線"  
},  
{  
    "Seq": 2,  
    "DateListed": "2013/09/01",  
    "Name": "花東縱谷單車人文旅行—鄉村、溫泉、部落",  
    "Ethnic": "阿美族、撒奇萊雅族、布農族、卑南族",  
    "Region": "花蓮縣、台東縣",  
    "Access": "",  
    "TravelItinerary": "第一天：花蓮火車站→德固灣撒奇萊雅文化廣  
    "TouristSeason": "",  
    "TravelDays": "三個太陽二個月亮",  
    "TravelCharacteristics": "花東縱谷線 如德固灣撒奇萊雅",  
    "TribeInformation": "",  
    "ImportantRite": "",  
    "Greetings": "",  
    "AssociationPopulation": "",  
    "CostPerPerson": "",  
    "Accommodation": "",  
    "Group": "2013年60條原住民族深度旅遊路線"  
},
```

Carrier 7:41 PM

# 原鄉旅遊路線

**老地方展新魅力—台東部落逍遙遊**  
一日遊：屏東縣→達仁鄉→大武鄉→多良向陽薪傳木工坊→金崙溫泉...

**花東縱谷單車人文旅行-鄉村、溫泉、部落**  
第一天：花蓮火車站→德固灣撒奇萊雅文化廣場、撒奇萊雅酒莊→吉...

**阿美族巴歌浪舞動之夜~水梯田海稻米、米粑流...**  
第一天：花蓮火車站→箭瑛大橋(1977年兩位老師的故事)→海岸山脈...

**南臺灣原鄉部落騎遇記(挑戰級)**  
第一天：高鐵左營站專車接送美濃人字山莊→109線道美濃自行車道...

**茂林紫斑蝶禮納里部落體驗二日遊**  
第一天：台北火車站→禮納里原住民風味餐→禮納里好茶部落（好茶...

**苗栗泰雅原鄉產業一日遊**  
一日遊：苗栗泰安鄉→司馬限市集→麥路豐市集→麻必浩市集→士林...

**烏來福山部落悠遊行~溪流德拉楠，巧遇馬告香**  
一日遊：烏來出發→五重溪瀑布→水圳步道散步趣→蝴蝶公園→部...

**探尋祕境禮納里部落二日遊**  
第一天：台北火車站→(中餐)禮納里風味餐→禮納里部落專人導覽→...

**渡邊吉海秧~大農大富平地森林單車遊**  
第一天：花蓮火車站→馬太鞍溪無毒農場→大農大富平地森林單車遊...

**嘎色鬧泰雅農夫~竹頭角木琴樂舞一日遊**  
一日遊：桃園縣復興鄉→桃園縣原住民文化會館→樂信瓦旦紀念公園...

**學習土地倫理~巒山森林博物館**  
第一天：海端鄉(布農族文物館)→拉瑪達星星雕塑像(布農族抗日英雄...

**來去烏來住一晚~文化，遇見繽紛泰雅(二日遊)**  
第一天：新店捷運站相見歡→內洞國家森林遊樂區→泰雅傳統美食→...

**烏來溫泉·野要獵人學野趣體驗一日遊**  
一日遊：新店捷運站→屈尺(泰雅遷移史)→泰雅博物館館內導覽→野...

**泰雅生態文化體驗(二日遊)**  
第一天：台北火車站→桃園火車站→桃園縣原住民文化會館→高義蘭...

# UI

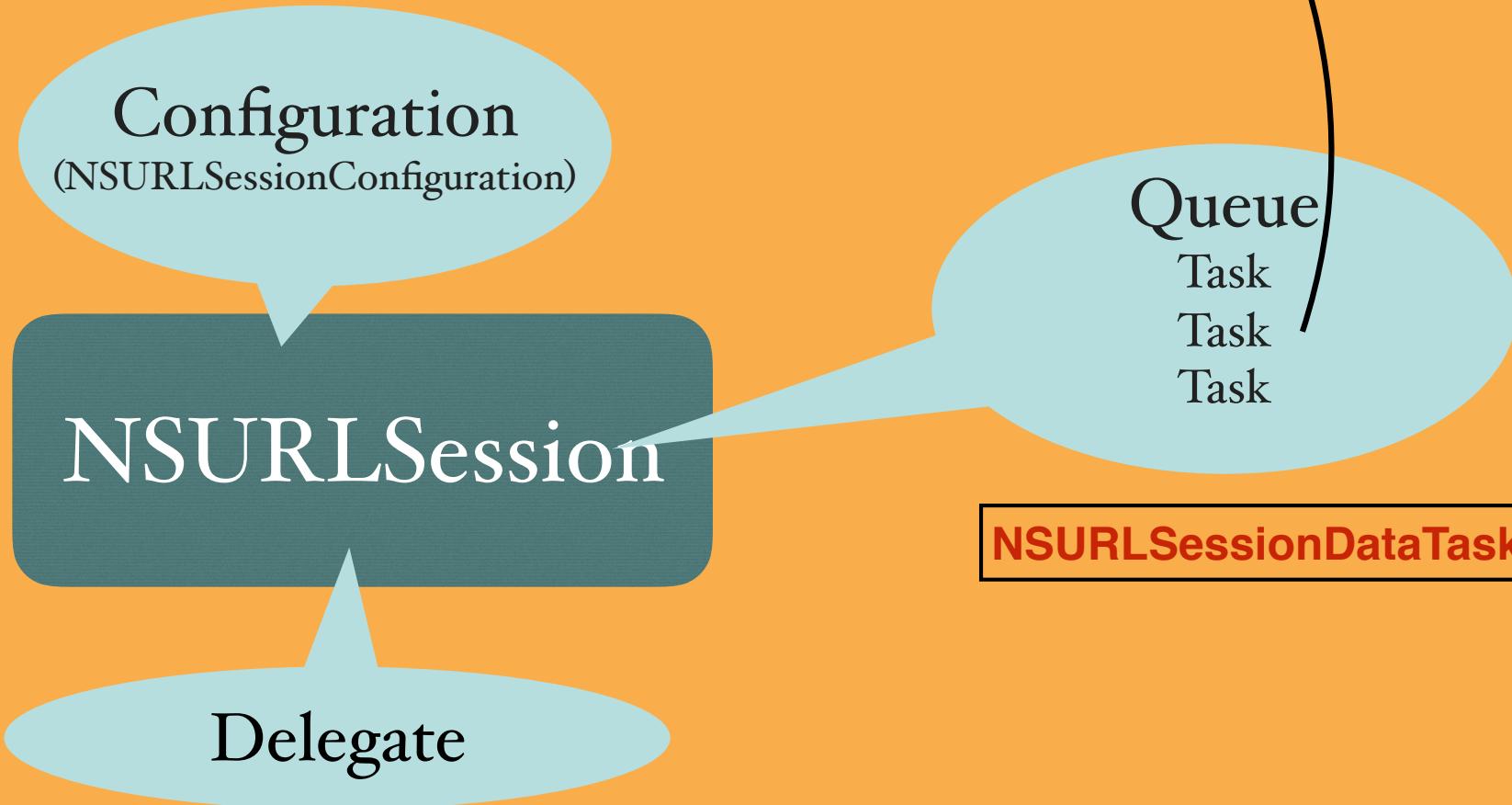
- New single view application
- UILabel
- UITableView
- UIActivityIndicatorView
- constraints and IBOutlets
- UITableViewDataSource, UITableViewDelegate
- UITableViewCell
- Add ATS



# ATS

Info.plist	M	Bundle OS Type code	String	APPL
Products		Bundle versions string, short	String	1.0
		Bundle creator OS Type code	String	????
		Bundle version	String	1
		Application requires iPhone environment	Boolean	YES
		Launch screen interface file base name	String	LaunchSc
		Main storyboard file base name	String	Main
		► Required device capabilities	Array	(1 item)
		► Supported interface orientations	Array	(3 items)
		► Supported interface orientations (iPad)	Array	(4 items)
		▼ App Transport Security Settings	Dictionary	(1 item)
		Allow Arbitrary Loads	Boolean	YES

```
defaultSessionConfiguration  
ephemeralSessionConfiguration
```



```
func URLSession(session: NSURLSession, task: NSURLSessionTask,  
didCompleteWithError error: NSError?)
```

```
func URLSession(session: NSURLSession, dataTask:  
NSURLSessionDataTask, didReceiveData data: NSData)
```

.....

# Data tasks

- \* `NSURLSessionDataTask`: send and receive data using `NSData` objects. Data tasks are intended for short, often interactive requests from your app to a server.

```
func URLSession(session: NSURLSession, dataTask:  
NSURLSessionDataTask, didReceiveData data: NSData)
```

```
func URLSession(session: NSURLSession, task:  
NSURLSessionTask, didCompleteWithError error: NSError?)
```

# Obtain the response data

- \* Provide a **completion handler block**. The URL loading class calls that block when it finishes receiving data from the server.
- \* Provide a custom **delegate**. The URL loading class periodically calls your delegate methods as it receives the data from the originating source. Your app is responsible for accumulating that data, if needed.

# data request

```
func fetchData() {  
  
    let session = NSURLSession(configuration:  
        NSURLSessionConfiguration.defaultSessionConfiguration())  
  
    let dataTask = session.dataTaskWithURL(NSURL(string:"http://  
lod2.apc.gov.tw/APCDataApi.ashx?apiId=102010&format=json")!)  
    { (data , response, error ) -> Void in  
  
        //檢查是否有錯誤  
        //將JSON資料轉成NSArray  
        //更新UITableView  
  
        session.finishTasksAndInvalidate()  
    }  
  
    busyIndicator.startAnimating()  
    dataTask.resume()  
}
```

# data request

```
//檢查是否有錯誤
if error != nil {
    print("request error= \(error)")
    session.finishTasksAndInvalidate()

    dispatch_async(dispatch_get_main_queue(), { () -> Void in
        self.busyIndicator.stopAnimating()
    })
}

return
}
```

# JSON

- \* NSJSONSerialization
- \* `func JSONObjectWithData(_ data: NSData, options opt: NSJSONReadingOptions) throws -> AnyObject`
- \* `func dataWithJSONObject(_ obj: AnyObject, options opt: NSJSONWritingOptions) throws -> NSData`
- \* `func isValidJSONObject(_ obj: AnyObject) -> Bool`

# data request

```
//將JSON資料轉成NSArray
do {
    let parseResult = try NSJSONSerialization.JSONObjectWithData
        (data! , options: NSJSONReadingOptions.AllowFragments)

    if parseResult is NSArray {
        self.datas = parseResult as? NSArray
    }else{
        print("Error, response data type error.")
        return
    }
    //更新UITableView
    dispatch_async(dispatch_get_main_queue(), { () -> Void in
        self.tableView.reloadData()
        self.busyIndicator.stopAnimating()
    })
}catch{
    print("Error parse json \(error )")
}
```

# UITableView DataSource

```
func tableView(tableView: UITableView, numberOfRowsInSection section: Int) -> Int {  
    if datas == nil {  
        return 0  
    }  
  
    return datas!.count  
}
```

# UITableView DataSource

```
func tableView(tableView: UITableView, cellForRowAtIndexPath indexPath: IndexPath) -> UITableViewCell {  
  
    let cell = tableView.  
        dequeueReusableCellWithIdentifier("dataCell")  
    if indexPath.row % 2 == 0 {  
        cell?.backgroundColor = UIColor.lightGrayColor()  
    }else {  
        cell?.backgroundColor = UIColor.whiteColor()  
    }  
  
    let data = datas![indexPath.row]  
    cell?.textLabel!.text = data["Name"] as? String  
    cell?.detailTextLabel!.text = data["TravelItinerary"] as?  
String  
  
    return cell!  
}
```

# Udacity - iOS Networking with Swift

- Lesson 0 : 網路基本概念
  - http request, http method, url, protocol....
- Lesson 1: Flicker
  - How to use flicker API
  - URL and parameters
  - create network request
- Lesson 2: Flicker
  - NSURLComponents
  - JSON Parse
  - Asynchronous Requests

# Udacity - iOS Networking with Swift

- Lesson 3: movie database
  - authentication flow
  - POST request
- Lesson 4: movie database
  - MVC
  - Refactor code
- Lesson 5:
  - Network condition test
  - Parse API
  - Facebook login

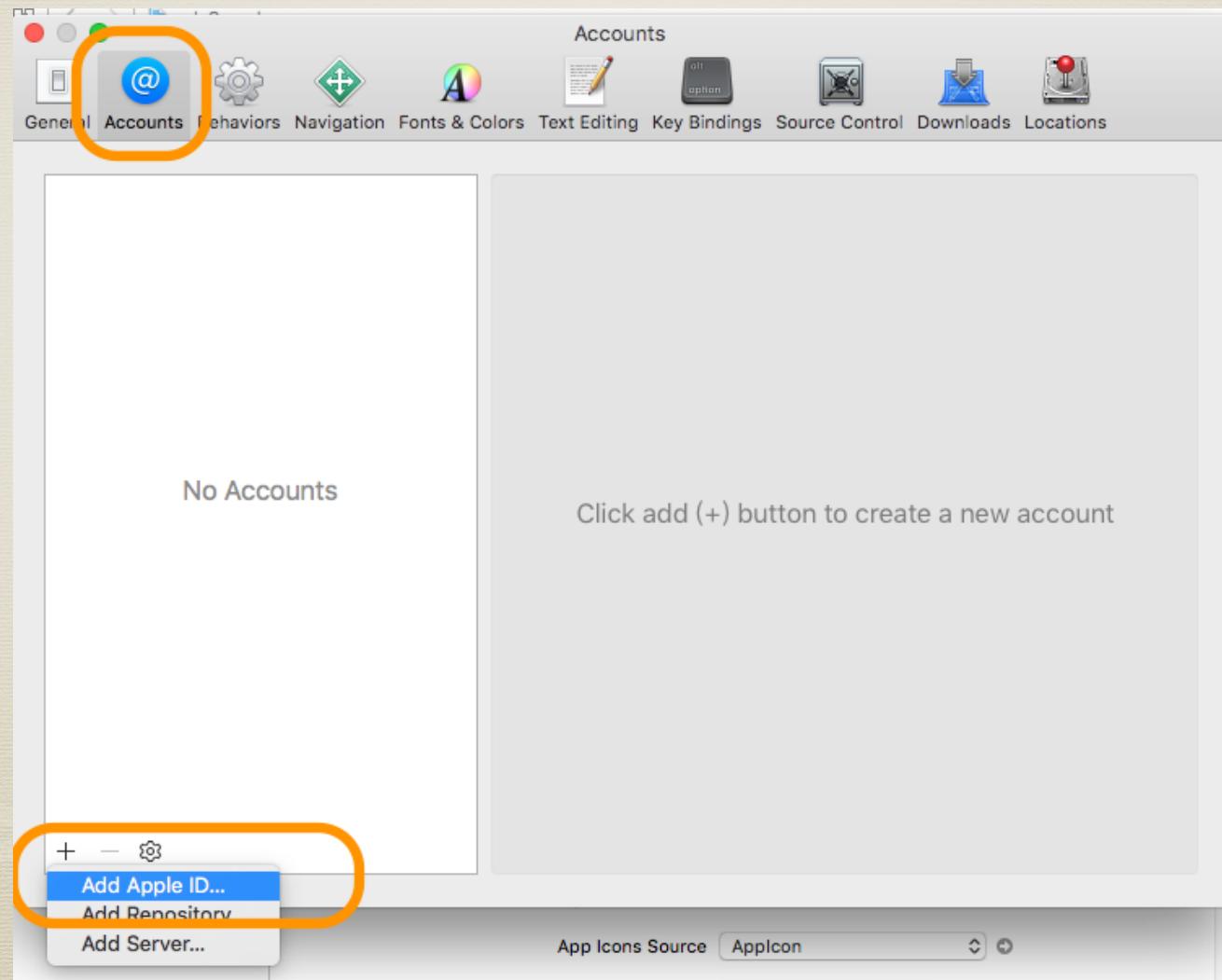
# Developer Account

- \* Apple Developer Program
  - \* Individual
  - \* Organization
- \* Apple Developer Enterprise Program
- \* Free developer account (Apple ID)

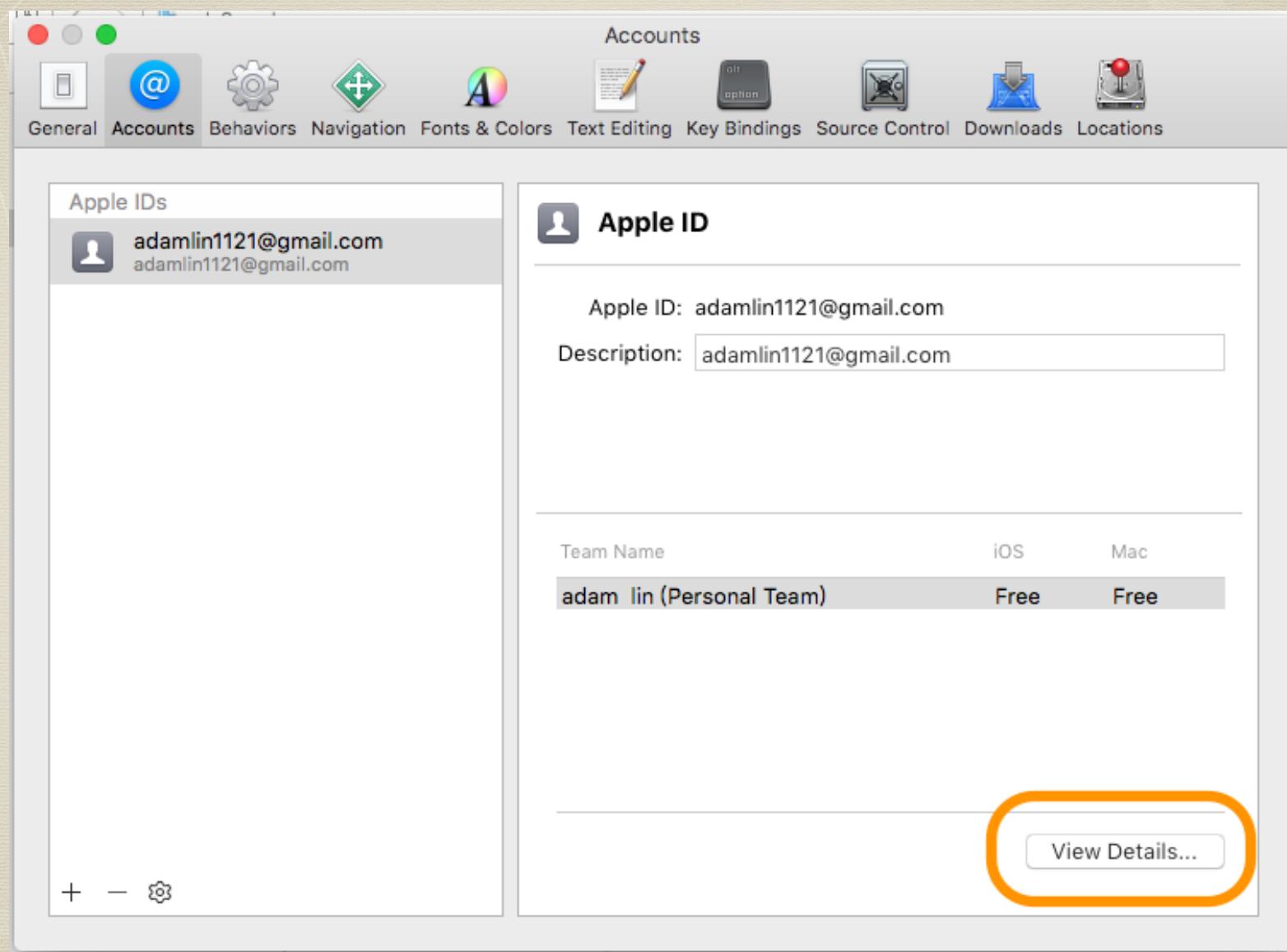
Free developer  
account (Apple ID)

# Demo

- \* Free developer account (Apple ID)









adam lin  
adamlin1121@gmail.com

[Signing Identities](#)

[iOS Development](#)

[iOS Distribution](#)

[Mac Development](#)

[Mac App Distribution](#)

[Mac Installer Distribution](#)

[Developer ID Application](#)

[Developer ID Installer](#)

Action

[Create](#)

[Create](#)

[Create](#)

[Create](#)

[Create](#)

[Create](#)

[Create](#)

[Provisioning Profiles](#)

Expires

Action

None

[Download All](#)

[Done](#)



adam lin  
adamlin1121@gmail.com

**Signing Identities**

- iOS Development
- iOS Distribution
- Mac Development
- Mac App Distribution
- Mac Installer Distribution
- Developer ID Application
- Developer ID Installer

Action

- Create

**Provisioning Profiles**

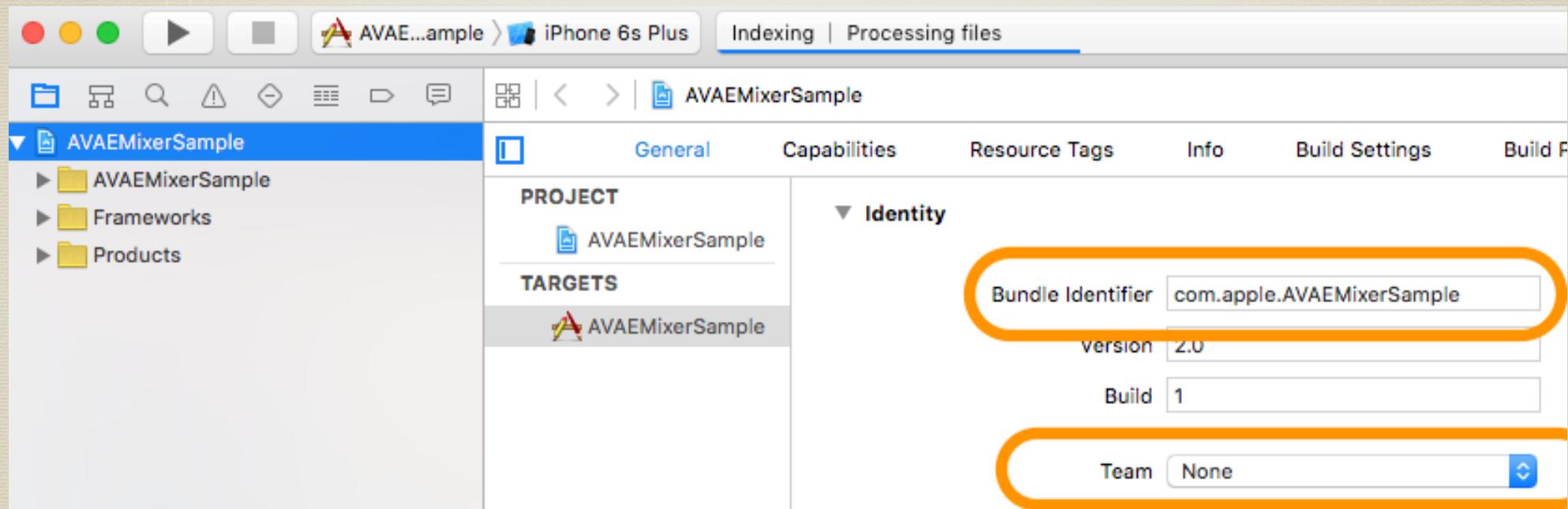
Expires

Action

None

Download All

Done



▼ Identity

Bundle Identifier

Version

Build

Team

 No matching provisioning profiles found

No provisioning profiles matching an applicable signing identity

were found.

[Fix Issue](#)



adam lin

adamlin1121@gmail.com

#### Signing Identities

iOS Development

iOS Distribution

Mac Development

Mac App Distribution

Mac Installer Distribution

Developer ID Application

Developer ID Installer

#### Action

Create

Create

Create

Create

#### Provisioning Profiles

#### Expires

#### Action

iOS Team Provisioning Profile: tw.adam.lin.SeismicXML

2016/4/11

Download All

Done

