**UIKit初探笔记**

應用程式開啟時，會自AppDelegate.swift開始，這隻檔案負責應用程式的生命週期，

像是啟動、閒置、進入後台、返回前台或是退出時要執行的動作。

接著看到 ViewController.swift ，是應用程式預設的主要視圖(View)控制器(Controller)，

所有需要的 UI 功能(像是按鈕、文字或圖案等等)，都必須在這個 ViewController 裡面建立，

通常會寫在viewDidLoad()這個方法裡面。

* 視圖(View)指的是在裝置螢幕上會呈現出來的東西，像是文字、圖片、元件(按鈕、選單、列表之類)等等。
* 視圖控制器(ViewController)則是負責資料的處理與編排設置視圖要如何呈現。
* 一個 UIViewController 負責一個畫面(即呈現視圖View)的功能，未來如果有多個畫面時，就需要各自建立一個不同的 UIViewController。

**1.文字標籤 UILabel：提示文字的原件**

**2.文字輸入 UITextField：输入文字的原件**

**3.輸入多行文字 UITextView**

**4.按钮UIButton**

如果要建立自定義的按鈕，則是使用UIButton(frame:)

例如：

// 使用 UIButton(frame:) 建立一個 UIButton

myButton = UIButton(frame: CGRect(x: 0, y: 0, width: 100, height: 30))

// 按鈕文字

myButton.setTitle("按我", forState: .Normal)

// 按鈕文字顏色

myButton.setTitleColor(UIColor.whiteColor(),forState: .Normal)

// 按鈕是否可以使用

myButton.enabled = true

// 按鈕背景顏色

myButton.backgroundColor = UIColor.darkGrayColor()

// 按鈕按下後的動作

myButton.addTarget(

self,

action: #selector(ViewController.clickButton),

forControlEvents: .TouchUpInside)

// 設置位置並加入畫面

myButton.center = CGPoint(

x: fullScreenSize.width \* 0.5,

y: fullScreenSize.height \* 0.5)

self.view.addSubview(myButton)

其中：

addTarget(target:, action:, forControlEvents:)用來負責按下按鈕後的動作，這是由 UIControl 所提供的方法，再繼承給 UIButton。這個方法的參數說明如下：

* target：當事件發生時，要呼叫哪一個物件。
* action：呼叫的物件要執行的方法，以#selector()來指定，ViewController.clickButton指的就是ViewController類別的clickButton方法。將ViewController省略掉，只填入clickButton也可以。
* forControlEvents：觸發的事件。(這裡則是**按下**事件。)

**5.提示框 UIAlertController**

UIAlertController(title:, message:, preferredStyle:)，是用來建立一個提示框的函式，參數如下：

title：提示的標題，會以粗體顯示。

message：提示的內容。

preferredStyle：提示框的類型，這邊填寫.Alert，會顯示在畫面中間，另外如果填寫.ActionSheet則是顯示在畫面底部。

接著可以看到用來建立提示框按鈕的UIAlertAction(title:, style:, handler:)，參數如下：

title：按鈕的文字。

style：按鈕的樣式，可選擇.Cancel、.Default、.Destructive等等。

handler：按下按鈕後要執行的動作，是一個型別為(action: UIAlertAction!) -> Void的閉包，如果不要有動作則是填入nil。

按鈕設置好之後，再使用alertController.addAction()方法，將按鈕加入到提示框控制器中。

最後則是對self使用presentViewController()方法來顯示提示框。

**6.圖片 UIImageView：顯示圖片主要是使用 UIImageView 及 UIImage 來完成**

**7.選取日期時間 UIDatePicker**

* datePickerMode：可以顯示的模式，有**只顯示時間**、**只顯示日期**、**顯示日期與時間**以及**倒數計時器**。
* minuteInterval：時間選取時的分鐘間隔，單位是分鐘。
* date：設置預設顯示的日期時間。
* minimumDate：可以選擇的最早日期時間。
* maximumDate：可以選擇的最晚日期時間。
* locale：顯示的語言環境。

**8.選擇器 UIPickerView**

UIPickerView 與 UIDatePicker 有點類似，但彈性更大，當你需要有一個或多個選擇項目時，可以設定自定義的選項及數目。

**9.開關 UISwitch**

**10.分段控制 UISegmentedControl**

**11.進度條 UIProgressView**

类型：長條進度條 UIProgressView 、 環狀進度條 UIActivityIndicatorView

NSTimer 是一個用來定時執行動作的類別。

scheduledTimerWithTimeInterval()方法的各參數：

* ti：第一個參數是間隔多久執行一次動作，單位是秒。
* target：執行動作的對象，通常是self。
* selector：定時執行的方法。
* userInfo：可以把需要的參數帶入方法中，沒有的話就填nil。
* repeats：是否可以重複執行，如果填false則是執行一次即不再動作。

**12.滑桿 UISlider：常用在控制音量或影片播放進度**

**13.步進器 UIStepper：可以用來為一個數值作增減的動作**

**14.網頁 UIWebView：應用程式中有時如果需要載入一些外部網站時，可以使用 UIWebView**

**15.表格 UITableView：當你需要將一批資料逐列顯示時常會使用到，每一個儲存格稱作一個 cell ，每個 cell 除了可以顯示文字外，還可以放置多個不同的元件。**

UITableView 委任的方法大多會有indexPath參數，這個參數有兩個屬性分別為section及row，是用來表示目前要設置的 cell 是屬於哪一組( section )的哪一列( row )，型別都為 Int 且都是由 0 開始算起。

**16.網格 UICollectionView：表現網格這樣多行多列元件的樣式， iPhone 內建的 照片 App 就是用 UICollectionView 為主要呈現方式。**

UICollectionView 有些地方與 UITableView 類似，它的每個儲存格也稱為 cell ，委任對象實作的方法也差不多。而 UICollectionView 有更多需要設置的地方，所以本節範例步驟較多，這邊先簡單講解一下：

1. 一開始需要先設置UICollectionViewFlowLayout，用來自定義呈現的樣式，再交給建立 UICollectionView 元件時的函式使用。
2. 接著建立 UICollectionView 元件，除了要註冊 cell 之外，如果要自定義每個 section 的 header 或 footer 時，也必須註冊 header 或 footer ，以供後續重複使用。
3. UICollectionView 的 cell 需要自定義一個繼承自 UICollectionViewCell 的類別，用來加上需要的元件，這邊會加上一張圖片( UIImageView )及一行字( UILabel )。
4. 最後要實作委任對象需要的方法。

**17.搜尋 UISearchController**

**18.滑動視圖 UIScrollView**

**19.多頁面:建立多個頁面，並在這些頁面中切換。**

* viewDidLoad()：最先被執行，時間點在 **View 被載入時**，不論切換退出這個頁面幾次，一個頁面只會執行一次viewDidLoad()。
* viewWillAppear()：在viewDidLoad()之後被執行，時間點在 **View 要被呈現前**，每次切換到這個頁面時都會執行。
* viewDidAppear()：在viewWillAppear()之後被執行，時間點在 **View 呈現後**，每次切換到這個頁面時都會執行。
* viewWillDisappear()：執行的時間點在 **View 要結束前**，每次要切換到別頁或是退出這個頁面時都會執行。
* viewDidDisappear()：執行的時間點在 **View 完全結束後**，每次要切換到別頁或是退出這個頁面時都會執行。

**20.導覽控制器 UINavigationController:更為方便的掌控頁面切換。**

**21.標籤列控制器 UITabBarController**

**22.手勢 UIGestureRecognizer:Tap 輕點、 Long Press 長按、 Swipe 滑動、 Pan 拖曳、 Pinch 縮放及Rotation 旋轉**

**23.簡單動畫 Animations**

* frame：視圖的位置與尺寸。
* bounds：視圖的尺寸。
* center：視圖的位置。
* alpha：視圖的透明度。
* backgroundColor：視圖的背景顏色。
* transform：平移、縮放或旋轉視圖。
* contentStretch：拉伸一部分的視圖。

**24.儲存資訊 NSUserDefaults:像是儲存使用者的開啟次數、使用時間或是有沒有使用過了哪些功能等等，這些少量資訊就可以使用 NSUserDefaults 來儲存。**