
UIView.animate 的 動畫功能

— Danny Shen —

UIKit 的動畫

UIKit iOS App 有很多實現動畫的方法, 較傳統的方法可使用UIView.animate 最簡單的寫法就是

```
//動畫開使前的UI
theView.backgroundColor = UIColor.red

UIView.animate(withDuration: [動畫秒數]) {

    self.theView.backgroundColor = UIColor(red: 1, green: 0, blue: 0, alpha: 0.4)

    //動畫結束時的UI, 包括位置角度顏色等
}
```

完整的 animate Function

```
class func animate(  
    withDuration duration: TimeInterval, //執行時間  
    delay: TimeInterval, //延遲時間  
    options: UIView.AnimationOptions = [], //各種選項 (見下一頁)  
    animations: @escaping () -> Void, //在執行後UIView 的目標顯示  
    completion: ((Bool) -> Void)? = nil //在執行完成後, 還要繼續執行的動作  
)
```

UIView.AnimationOptions

這兒的選項，主要是動畫的速度曲線與重複等。預設 值是線性，不重覆，但有了這些參數後，可以執行較複雜的動畫，如

`.repeat` 重覆動畫

`.autoreverse` (配合重覆) 完成動畫後恢復原狀

`.curveEaseInOut` 指定緩入緩出曲線，使動畫緩慢開始，在持續時間中間加速，然後在完成前再次減速。

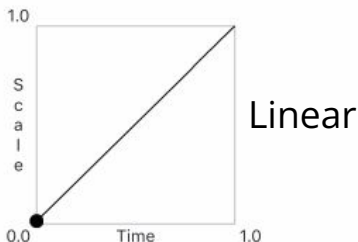
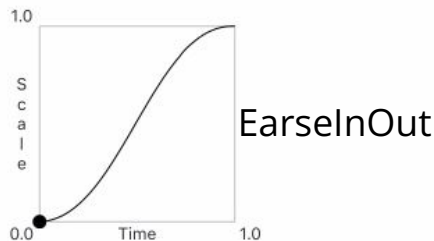
`.curveLinear` 指定線性動畫曲線使動畫在其持續時間 內均勻發生。

更多曲線說明：

<https://blog.csdn.net/gang544043963/article/details/80380436>

還有許多參數，可參考線上文件 說明：

<https://developer.apple.com/documentation/uikit/uiview/animationoptions>



一個閃動的 UIView 動畫

假設要設定的 UIView 已放入變數 theView 中，並已顯示畫面中，要它閃動紅色，且平順的改變顏色，就可以：

```
theView.backgroundColor = UIColor.red
```

```
UIView.animate(withDuration: 1,
```

```
    delay: 0,
```

```
    options: [.curveEaseInOut,.repeat,.autoreverse]) {
```

```
    self.theView.backgroundColor = UIColor(red: 1, green: 0, blue: 0, alpha: 0.4)
```

```
}
```

```
//=====
```

```
//theView.layer.removeAllAnimations() // 停止動畫，若動畫有多個 UIView 時，要會部移除
```

另一種新版動畫 UIViewPropertyAnimator

使用 UIViewPropertyAnimator 也可以做類似的動畫，例如之前的動畫也可以這樣寫

```
theView.backgroundColor = UIColor.red
```

```
UIViewPropertyAnimator.runningPropertyAnimator(withDuration: 3, delay: 0) {  
    self.theView.backgroundColor = UIColor(red: 1, green: 0, blue: 0, alpha: 0.4)  
}
```

基本上，它和 UIView.animate 的原理方法相近，但如其名，提供了更多的參數選擇，也可能做更多的事。實際上，它是為了做一些互動式的動畫，例如做一些點取的畫面位置或點取的 UIView 相關的動畫