# Sequence 应用（Tweener的链表）

#### 基本介绍

a.序列和动画一样，但是并不是动画属性或是其他什么值而是作为动画一个群体存在。

b.序列可以无层数限制地被嵌套在任意序列中。

c.序列不需要一个接一个延续，你可以将任意动画插入到序列中。

d.一个动画只能用于某一个序列中，意味着不能在多个序列中插入相同动画。

e.得到主序列的时候，你不能单独控制其中被嵌套的动画或序列。

#### 常用方法：

**DOTween.Sequence()**  静态,返回一个新的空序列。

**sequence.Append(Tween tween)** 添加一个动画到序列末尾。**sequence.AppendCallback(TweenCallback cb)** 添加回调函数到序列末尾。**sequence.AppeedInterval(float interval)** 添加一段空时间到序列末尾。(当序列前面动画播放完毕时会延迟一段时间)

**sequenve.Insert(float time,Tween tween)** 插入一段动画到指定时间。(注意动画是和原序列当前时间动画合并重叠)

**sequenve.InsertCallback(float time,TweenCallback cb)** 插入回调函数到序列指定时间。

**sequenve.Join(Tween tween)**  插入动画与前一个动画(这里的最后是指最后加入序列而非序列末尾)同时播放。

**sequenve.Prepend、sequenve.PrependCallback、sequenve.PrependInterval 和前面三个类似，这里是加入到序列开头**

#### 飘字效果示例要求：

a.字体向上移动同时颜色Alpha从0渐变到1。

b.延迟1秒，字体显示一秒钟。

c.字体向上移动同时颜色Alpha从1渐变到0。

#### 飘字动画效果代码示例

1. **using** UnityEngine;
2. **using** System.Collections;
3. **using** UnityEngine.UI;
4. **using** DG.Tweening;
5. **public** **class** FlyText : MonoBehaviour {
7. **public** **void** FlyTo(Graphic graphic)
8. {
9. RectTransform rt = graphic.rectTransform;
10. Color c = graphic.color;
11. c.a = 0;
12. graphic.color = c;                                                   //先将字体透明
13. Sequence mySequence = DOTween.Sequence();                            //创建空序列
14. Tweener move1 = rt.DOMoveY(rt.position.y + 20, 0.5f);            //创建向上移动的第一个动画
15. Tweener move2 = rt.DOMoveY(rt.position.y + 40, 0.5f);           //创建向上移动的第二个动画
16. Tweener alpha1 = graphic.DOColor(**new** Color(c.r, c.g, c.b, 1), 0.5f);//创建Alpha由0到1渐变的动画
17. Tweener alpha2 = graphic.DOColor(**new** Color(c.r, c.g, c.b, 0), 0.5f);//创建Alpha由1到0渐变的动画
18. mySequence.Append(move1);                  //先添加向上移动的动画
19. mySequence.Join(alpha1);                   //同时执行Alpha由0到1渐变的动画
20. mySequence.AppendInterval(1);              //延迟1秒钟
21. mySequence.Append(move2);                  //添加向上移动的动画
22. mySequence.Join(alpha2);                   //同时执行Alpha由1到0渐变的动画
23. }
24. **public** **void** Update()
25. {
26. **if**(Input.GetMouseButtonDown(0))
27. {
28. FlyTo(GetComponent<Text>());
29. }
30. }
31. }

# Path 应用