

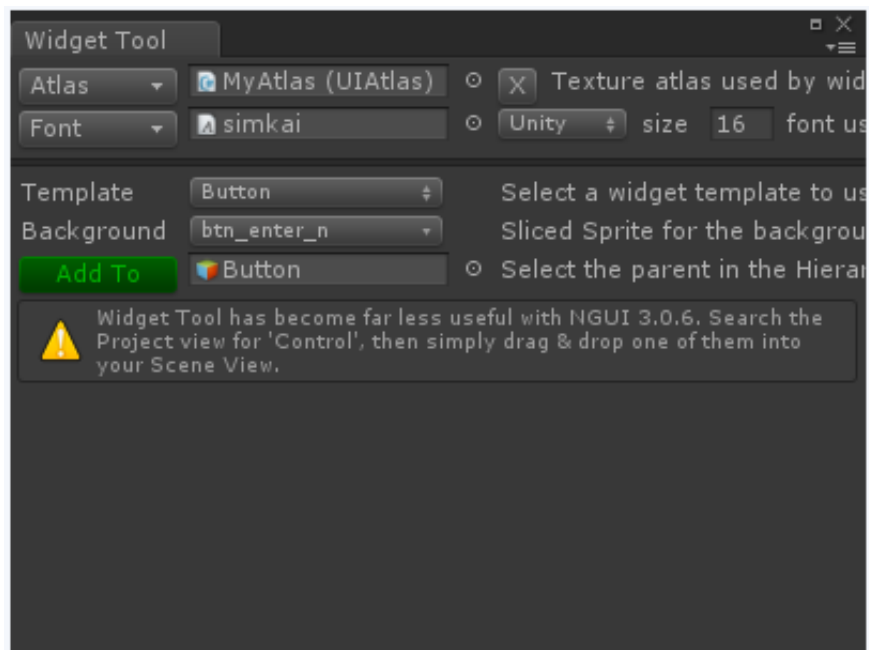
## UIButton 按钮

默认接受点击事件

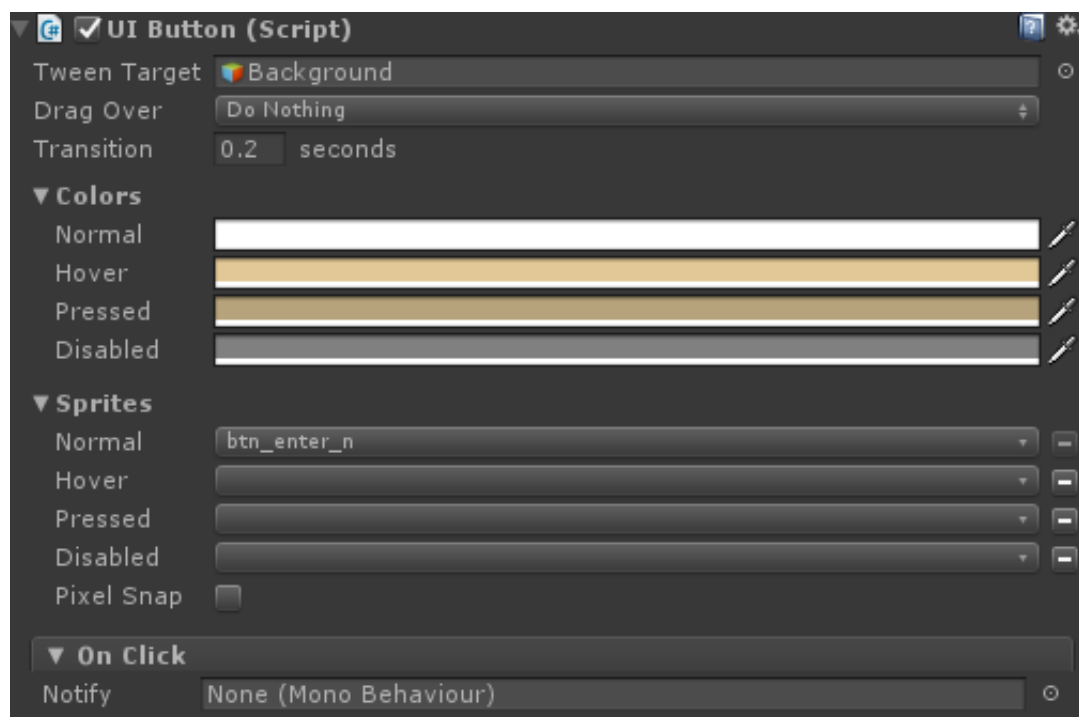
由一个Sprite和Label组成

创建方法：NGUI-OPEN-WidgetTool

选择对应的图集和字体



## UIButton组件



Target 点击效果作用的对象

Transition 效果过渡的持续时间

Colors 按钮对应在不同状态下的颜色

Sprites 按钮对应在不同状态下的显示精灵

OnClick 按钮按下时 触发的事件回调

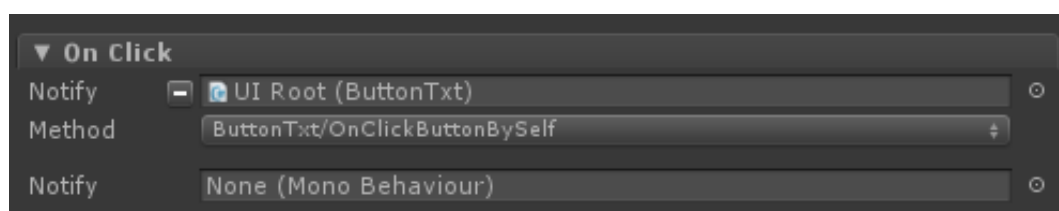
## 1 最简单的回调方式：

脚本和UIButton绑定在同一个物体上

```
1 public class ButtonTxt : MonoBehaviour {  
2  
3     void OnClick()  
4     {  
5         Debug.Log("按钮按下");  
6     }  
7 }  
8
```

## 2 设置Notify 控制按钮回调

```
1 public class ButtonTxt : MonoBehaviour {  
2  
3     //格式: public void name(void)  
4     public void OnClickButtonBySelf()  
5     {  
6         Debug.Log("按钮按下");  
7     }  
8 }
```



## 3 UIButtonMessage

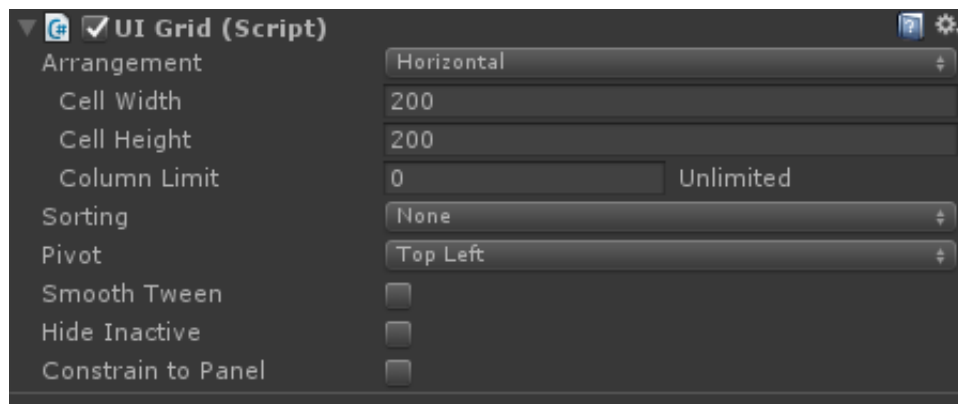
## 4 委托监听

```
1 public class ButtonTxt : MonoBehaviour {
2
3     public UIButton button;//按钮
4     public UILabel label;//按钮上的文本标签
5
6     void Start()
7     {
8         label.text = "醒醒";
9         //使用EventListener设置回调函数
10        UIEventListener.Get(button.gameObject).onClick =
OnClickButton;
11    }
12
13    void OnClickButton(GameObject btn)
14    {
15        //打印按钮物体的名字
16        Debug.Log(btn.name);
17    }
18 }
19
```

## 设置按钮的不可用状态

```
1 public class ButtonTxt : MonoBehaviour {
2
3     public UIButton button;
4
5     void Start()
6     {
7         //NGUI的设置不可用的代码 调整按钮的颜色 并且 不享受点击事件
8         button.isEnabled = false;
9         //是Unity自带的控制是否可用的方法 会把UIButton组件设置为不可用
10        //button.enabled = false;
11    }
12 }
```

## UIGrid



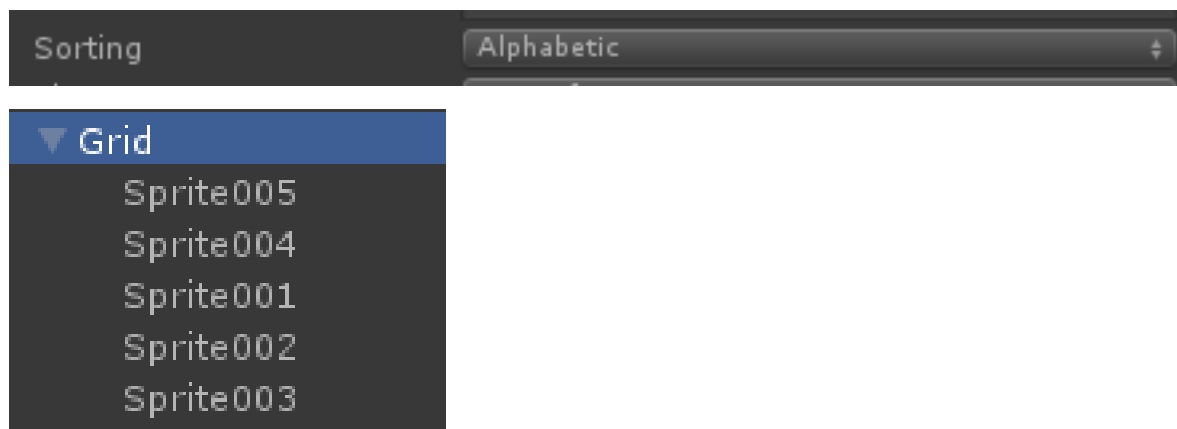
ArrangeMent 排列方式 水平和垂直

CellWidth CellHeight 决定所有条目之间的间隔大小

RowLimit 决定了排列的最大行数或者列数

## Sorting 排序

排序模式 根据名字编号来排序

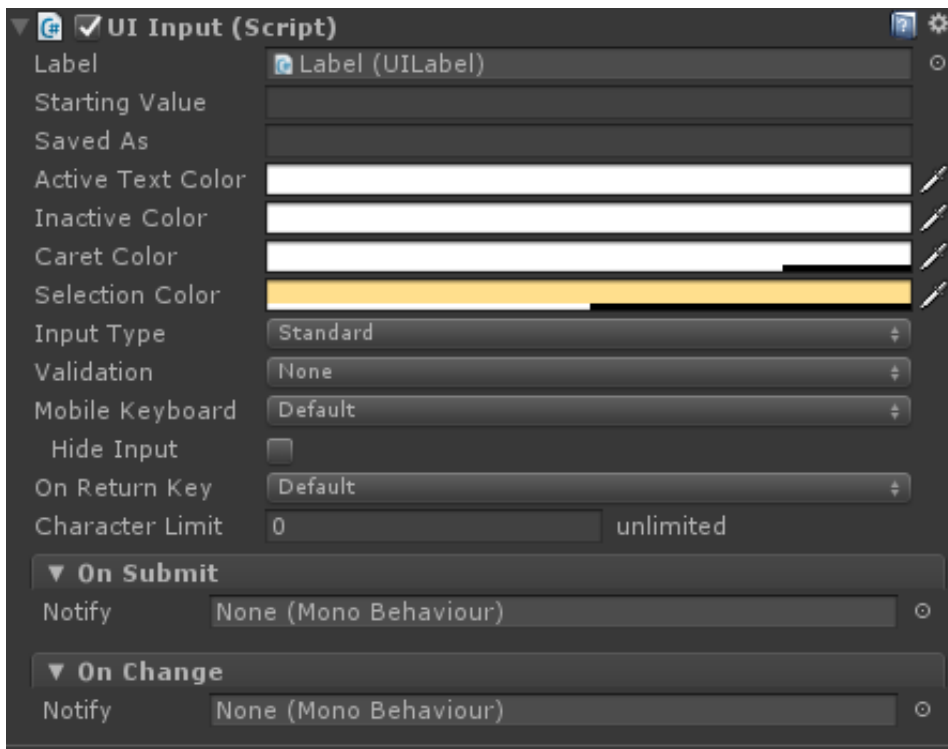


SmoothTween 动画平滑过渡效果

Hide InActive 是否隐藏未被激活的对象

Constrain in Panel 保持在Panel范围内显示 超过部分隐藏

## UIInput



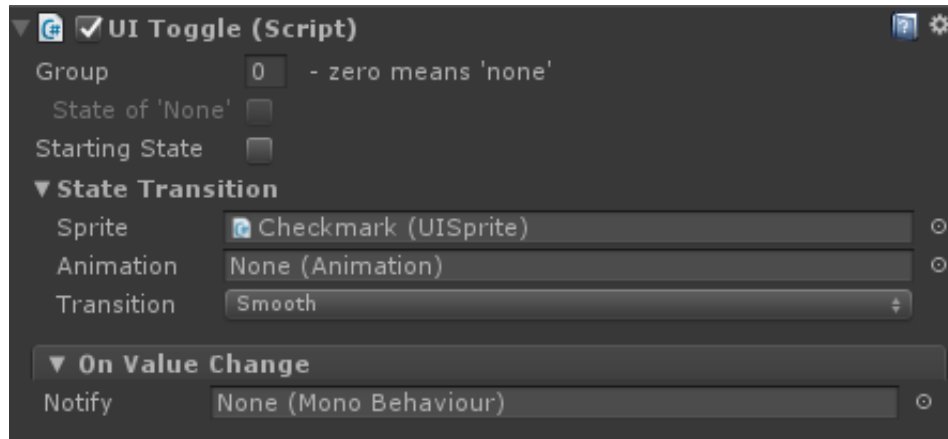
**StartingValue** 输入框在运行开始时显示的文字 在Label里设置也可以

运行时 如果没有StartingValue会使用Label的默认字符串 如果有使用StartingValue

**Saved As** 保存输入内容到PlayerPrefs SavedAs的key就是组件中输入的字符串

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class ButtonTxt : MonoBehaviour {
6
7     public UIInput input;
8
9     void Start()
10    {
11        input.onSubmit.Add(new EventDelegate(this, "OnInputSucess"));
12    }
13
14    public void OnInputSucess()
15    {
16        Debug.Log(input.value);
17    }
18
19 }
```

## UIToggle 开关

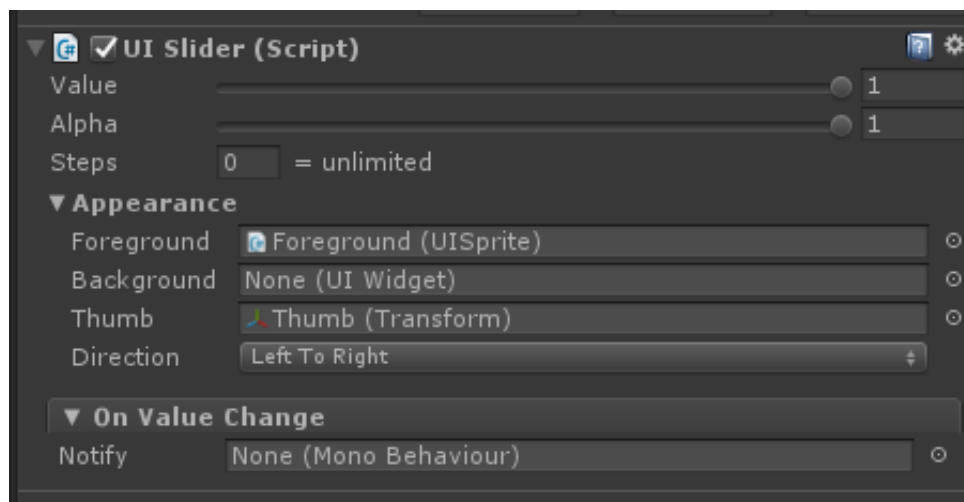


Group 用来实现互斥功能 比如制作类似选择题的效果 0代表不在Group范围内

StartingState 默认起始状态 是否被勾选



## UISlider和ProgressBar 拖动条 进度条



Value 取值范围0-1 拖动条可以手动控制 进度条代码控制

Direction 方向 左到右 上到下

UIPopupList 弹出框 菜单项

