

# LightUI

## 功能介绍

LightUI 是一个轻量级的UI组件，可以使用更为易读的方式创建Mod的GUI界面。

在使用时，首先需要创建一个LightUI 的实例，例如

```
public LightUI.LightUI guiLayout = new LightUI.LightUI();
```

在BIE MOD内，通过 `YanLib.ModHelper.ModHelper` 来设置图形界面时，需要对 `ModHelper.SettingUI` 进行赋值来设置图形界面的内容。

由于 `UnityUIKit` 和 `TaiwuUIKit` 的开发者更倾向于使用大括号初始化来构造UI界面，但是，由于UMM MOD原有的GUI是通过 `GUILayout` 类来创建的，因此迁移时会造成一定的障碍。

而LightUI则是对 `TaiwuUIKit` 进行封装，从而更符合原有的习惯。

提供的方法有如下：

- `BeginVertical` 开始创建竖直布局
- `EndVertical` 结束竖直布局创建，返回布局元素
- `BeginHorizontal` 开始创建水平布局
- `EndHorizontal` 结束水平布局创建，返回布局元素
- `Label` 在布局内添加一个标签
- `Toggle` 在布局内添加一个单选框
- `Slider` 在布局内添加一个滑条
- `Button` 在布局内添加一个按钮
- `InputField` 在布局内添加一个文本编辑框
- `ToggleGroup` 在布局内添加一个单选组（多个单选框只能同时选中一个）
- `Space` 在布局内添加一个空格
- `Find`
  - 根据ID查找元素
  - 根据元素类型获取该类型的元素集合

一个对 `SettingUI` 赋值的简单例子如下：

```
guiLayout.BeginVertical();
guiLayout.Label("Hello world!");

Mod.SettingUI = guiLayout.EndVertical();
```

## 与 GUILayout 的区别

`GUILayout` 在每次更新GUI的时候都会调用一次，而BIE框架中，GUI在 `Awake()` 函数中进行创建，在 `Update()` 函数中进行状态更新。

如果想要让某个组件在创建时隐藏，可以在创建时传入参数 `defaultActive`，而水平、竖直布局若要隐藏，需要在BeginXXX的时候传入参数。

如果想要改变某个组件的隐藏状态，可以在 `Update` 函数内，使用 `SetActive(bool)` 函数。