

简单容器(Group)

在 EUI 提供的容器中，Group 是最轻量级的，它本身不可以设置皮肤，也不具备外观，它的呈现只取决于内部的显示对象。如果需要使用容器，并且没有设置皮肤的需求，那么请尽量使用 Group。

Group 操作内部对象的方法和 Sprite 基本一致，只是增加了2个方法 `getElementAt()` 和 `numElements()`，在后面的 数据容器 (../../extension/EUI/dataCollection/dataGroup/index.html) 章节将会详细说明。

如果自定义一个类，继承自Group，那么其内部的其它对象应该在 `createChildren()` 方法中创建和添加，也就是说，要覆盖 Group 的 `createChildren()` 方法。参见下面的例子：

```
class GroupDemo extends eui.Group {
    constructor() {
        super();
    }
    protected createChildren(): void {
        super.createChildren();
        var btn: eui.Button = new eui.Button();
        btn.label = "Button";
        this.addChild(btn);
    }
}
```

eui 中容器的一个显著特点是，可以配置 layout 对象，来实现不同的布局方式。这对开发工作是非常有好处的，可以使用默认的几个布局类，来节省大量的编码工作。布局的详细内容，请看 使用布局类 (../../extension/EUI/useLayout/basiclayout/index.html)。下面的示例演示了使用垂直布局来排列4个按钮:

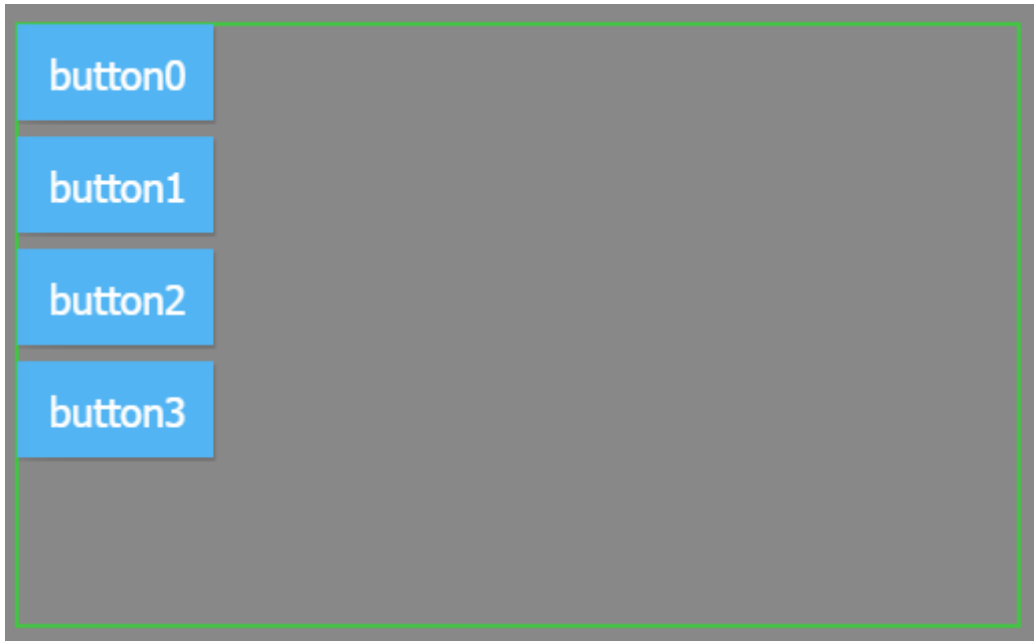
```

/**
 * Created by FCX on 1/14/2016.
 */
class GroupDemo extends eui.Group {
    constructor() {
        super();
    }
    protected createChildren():void {
        super.createChildren();
        this.layoutContents();
    }
    private myGroup:eui.Group;
    private layoutContents():void {
        // 设置默认主题
        var theme = new eui.Theme(`resource/default.thm.json`, this.stage);
        // 创建一个 Group
        var myGroup = new eui.Group();
        myGroup.x = 100;
        myGroup.y = 100;
        myGroup.width = 500;
        myGroup.height = 300;
        this.myGroup = myGroup;
        this.addChild(myGroup);
        // 绘制矩形用于显示 myGroup 的轮廓
        var outline:egret.Shape = new egret.Shape;
        outline.graphics.lineStyle(3,0x00ff00);
        outline.graphics.beginFill(0x1122cc,0);
        outline.graphics.drawRect(0, 0, 500, 300);
        outline.graphics.endFill();
        myGroup.addChild(outline);
        // 在 myGroup 中创建4个按钮
        for (var i:number = 0; i < 4; i++) {
            var btn:eui.Button = new eui.Button();
            btn.label = "button" + i;
            btn.x = 10 + i * 30;
            btn.y = 10 + i * 30;
            myGroup.addChild(btn);
        }
        // 使用绝对布局，会忽略 myGroup 中按钮的自定义坐标
        myGroup.layout = new eui.VerticalLayout();
    }
}

```

```
}
```

编译运行，效果如图：



一些使用技巧：

- 调用 `removeChildren()` 方法可以删除所有的内部显示对象
- Group 和所有其他 UI 组件都遵循一个原则：组件在没被外部显式设置尺寸(直接设置 width/height)的前提下，会自己测量出一个“合适”的大小。这时候 Group 宽高就是 `contentWidth` 和 `contentHeight` 的宽高。如果显式设置了 Group 的尺寸，则它的尺寸不一定等于内部对象尺寸。比如 Group 高度是 100 像素，但内部几个按钮的高度加起来是 400 像素，此时通过 `group.height` 获取的高度还是 100。您可以使用 `contentWidth` 和 `contentHeight` 属性来获取内部高度。
- 如果内容尺寸超出容器尺寸，默认是全部显示的，您可以设置 `scrollEnabled = true`，这样超出的部分就不再显示了。

效果如下图：

