StackView学习

自适应、适配、布局这几个关键词一直伴随着iOS开发，从以前的单一尺寸屏幕，到现在的多尺寸屏幕，Apple一直致力于让开发人员尽可能少在这些事上耗费过多的精力，所以Apple在2012年推出了Auto Layout特性，2014年又推出了Adaptive Layout、Size Classes，2015年又推出了Stack View。这些无一不是我们开发者做适配的利器。今天就让我们看看StackView是怎么一回事。

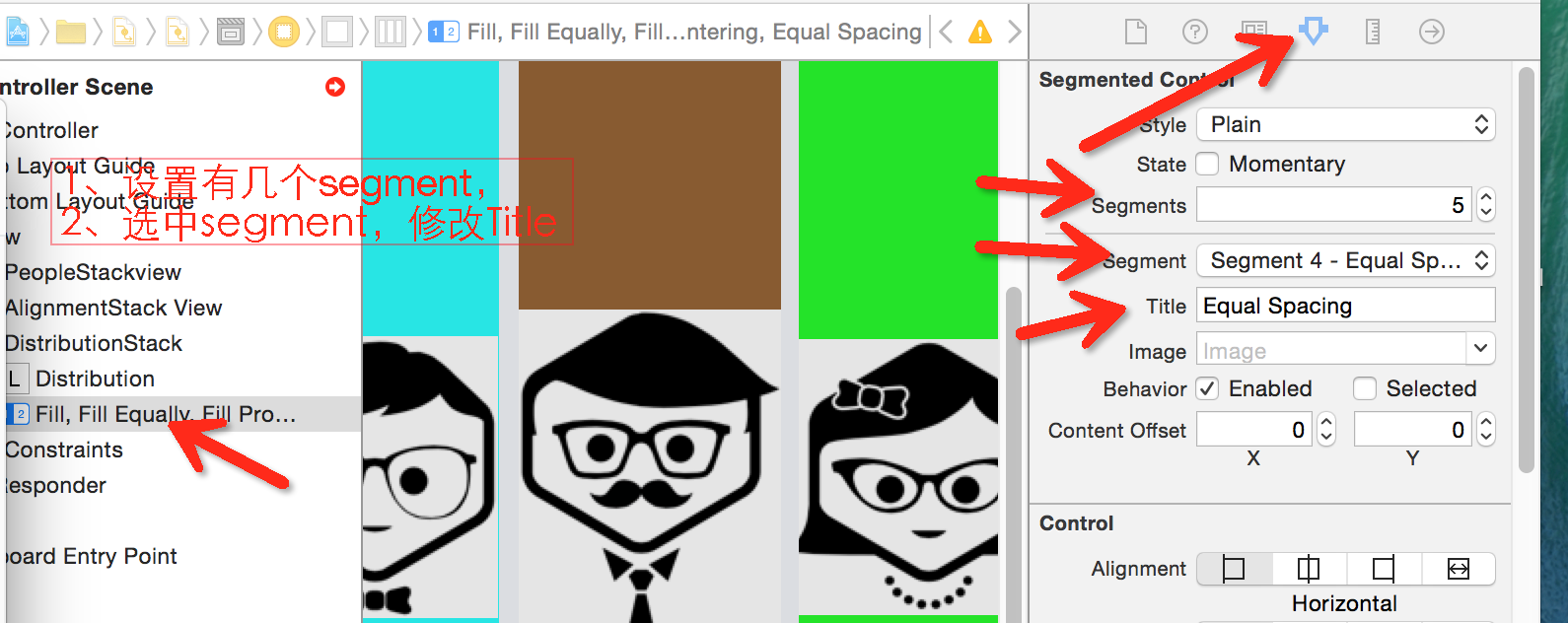
在 iOS9 中，Apple 引入了 UIStackView，他让你的应用可以通过简单的方式，纵向或横向的叠放你的 views。UIStackView 采用 auto layout 的方式来管理他的子视图的位置和尺寸。让你更简单的构建自适应的 UI。

Apple对于布局的发展历程大致的介绍到此，下面结合实例对StackView的属性进行学习；

StackView其实一个视图容器，不过它会对它的子视图根据一定规则自动布局，将子视图按栈的排列方式进行布局，并且有几个主要的属性：

新建工程，依旧用storyboard进行操作；

1. 选中storyboard，在页面搜索Stack View，你会发现StackView有水平和垂直两个方向的布局模式（内部空间的布局规则）；
2. 拖拽一个新的水平StackView（PeopleStackView）并设置它在父视图的位置，相对位置（0、100、0、0）🡪《上、下、左、右》
3. 拖进stackview四张图片，默认对齐方式为Fill；
4. StackView常用两个属性，Aligntioin/Distributon;先用代码实现，为方便起见，在PeopleStackView下方拉进一个新的水平StackView（AligntionStack）并设置它在父视图的位置，里边放一个Label，一个segment，选中segment，修改属性，



1. 同理设置Distributon的Stackview，以及内部控件；
2. 对齐方式

// Fill：子视图填充StackView。

// Leading：靠左对齐。

// Trailing：靠右对齐。

// Center：子视图以中线为基准对齐。

// Top：靠顶部对齐。

// Bottom：靠底部对齐。

// First Baseline：按照第一个子视图中文字的第一行对齐。

// Last Baseline：按照最后一个子视图中文字的最后一行对齐。

1. 分布比例

// Fill：默认分布方式。

// Fill Equally：子视图的高度或宽度保持一致。

// Fill：Proportionally：StackView自己计算出它认为合适的分布方式。

// Equal Centering：每个子视图中心线之间保持一致的分布方式。

// Equal Spacing：子视图保持同等间隔的分布方式。

说明：

1、不论你拉取的是水平还是垂直可以在属性中更改，

2、这些属性都可以可视化更改，故不再累述

3、stackview就是一个容器，会根据你设置的对齐方式和分布方式内部实现布局。