**第十三讲ionic js指令布局介绍**

学习要点：

1. 标题栏 : ion-header-bar

2. 页脚栏 : ion-footer-bar

3. header/footer : 样式及内容

4. 内容区 : ion-content

5. 滚动框 : ion-scroll

6. 下拉更新数据 : ion-refresher

7. 上拉更新数据 : ion-infinite-scroll

8. 脚本接口 : $ionicScrollDelegate

1. **标题栏 : ion-header-bar**

**ion-header-bar**指令声明一个标题栏元素，标题栏总是位于屏幕的顶部：

**<ion-header-bar>...</ion-header-bar>**

**ion-header-bar指令有两个可选的属性：**

**align-title** - 设置标题文字的对齐方式。允许值：left | right | center，分别对应左对齐、 右对齐和居中对齐。

* **no-tap-scroll** - 当点击标题时，是否将内容区域自动滚动到最开始。允许值：true | false，默认为false。

**用法：**

|  |
| --- |
| <ion-header-bar align-title="left" class="bar-positive">  <div class="buttons">  <button class="button" ng-click="doSomething()">左侧按钮</button>  </div>  <h1 class="title">Title!</h1>  <div class="buttons">  <button class="button">右侧按钮</button>  </div>  </ion-header-bar>  <ion-content>  一些内容!  </ion-content> |

1. **页脚栏 : ion-footer-bar**

**ion-footer-bar指令声明一个页脚栏元素，页脚栏总是位于屏幕的底部：**

**<ion-footer-bar>...</ion-footer-bar>**

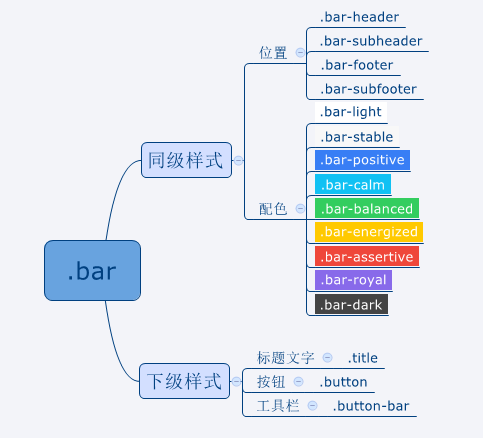
**ion-footer-bar指令有一个可选的属性：**

**align-title - 设置标题文本的对齐方式。允许值：left | right | center 。 现在好像没有作用**

|  |
| --- |
| <ion-content>  一些内容!  </ion-content>  <ion-footer-bar align-title="left" class="bar-assertive">  <div class="buttons">  <button class="button">左侧按钮</button>  </div>  <h1 class="title">Title!</h1>  <div class="buttons" ng-click="doSomething()">  <button class="button">右侧按钮</button>  </div>  </ion-footer-bar> |

1. **header/footer : 样式及内容**

ion-header-bar和ion-footer-bar经过编译后其样式类将分别被设置为.bar.bar-header 和.bar.bar-footer，回忆下我们在CSS框架课程中已经了解到的内容：



显然，你可以使用这些样式调整ion-header-bar/ion-footer-bar的外观！

1. **内容区 : ion-content**

**使用ion-content指令定义内容区域：**

<ion-content>...</ion-content>

ion-content占据header和footer以外的剩余区域。当内容超过可视区域时，ion-content 可以滚动以显示被隐藏的部分。

试着滚动右边示例效果的内容区域，你会发现浮现的滚动条。当滚动停止时，浮动条消失。 这是ionic定制的滚动视图，可以使用**overflow-scroll**属性设置**使用系统内置的滚动条**：

<ion-content **overflow-scroll="true"**>...</ion-content>

**scroll="false" 禁止滚动 用于百度地图 或者谷歌地图显示**

<ion-content **scroll="false"** >...</ion-content>

1. **ionic滚动框 : ion-scroll**

**ion-scroll**指令声明一个可滚动的容器元素，用户可以按住内容进行拖动：

<ion-scroll>

<!--content-->

</ion-scroll>

**ion-scroll指令有两个常用的可选属性：**

* **direction** - 内容可以滚动的方向。允许值：x|y|xy。默认为 y。
* **zooming** - 是否支持pinch-to-zoom(捏拉缩放)。允许值：true | false。

在使用ion-scroll时，需要显式指定滚动框元素及内容元素 的大小（高度和宽度）：



**6.** **下拉更新数据 : ion-refresher**

使用指令**ion-refresher**可以为滚动容器（ion-scroll或ion-content）增加 拉动刷新/pull-to-refresh的功能：

**<ion-refresher></ion-refresher>**

**ion-refresher指令有以下可选的属性：**

* **on-refresh** - 当用户向下拉动足够的距离并松开时，执行此表达式
* **on-pulling** - 当用户开始向下拉动时，执行此表达式
* **pulling-text** - 当用户向下拉动时，显示此文本
* **pulling-icon** - 当用户向下拉动时，显示此图标
* **refreshing-icon** - 当用户向下拉动并松开后，显示的等待图标。ionic推荐使用spinner 代替这个属性
* **spinner - 和refreshing-icon**的作用一样，但spinner是基于SVG的动画
* **disable-pulling-rotation** - 禁止下拉图标旋转动画

**注意在刷新完毕后，应当使用作用域的$broadcast()方法通知框架**：

$scope.$broadcast("scoll.refreshComplete")

**spinner使用：**

|  |
| --- |
| <ion-infinite-scroll on-infinite="load\_more();" **spinner**="android">  </ion-infinite-scroll> |

<http://www.ionicframework.com/docs/api/directive/ionSpinner/>

**7.** **上拉更新数据 : ion-infinite-scroll**

使用**ion-infinite-scroll**指令可以为滚动容器（ion-scroll或ion-content）增加 滚动刷新功能：

**<ion-infinite-scroll on-infinite="">...</ion-infinite-scroll>**

**ion-infinite-scroll指令有如下属性：**

* **on-infinite** - 必须。当滚动到底部时执行此表达式
* **distance** - 可选。距底部距离百分比。当距离底部超过此数值时，执行on-infinite。默认为1%
* **icon** - 可选。载入时显示的图标。默认是**ion-load-d**。ionic推荐使用spinner代替icon属性
* **spinner** - 可选。载入时的spinner。默认是ionSpinner
* **immediate-check** - 可选。是否在载入时立即检查滚动框范围

载入图标使用icon属性设置为ion-load-a试试

**8.** **$ionicScrollDelegate 指令**

可以使用服务**$ionicScrollDelegate**，通过脚本控制滚动容器（ion-scroll或ion-content）。 $ionicScrollDelegate服务提供的常用方法如下：

* **resize()**

重新计算容器尺寸。当父元素大小变化时，应当调用此方法

* **scrollTop([shouldAnimate])**

滚动到内容顶部。shouldAnimate参数为true|false，表示是否使用动画展示滚动过程

* **scrollBottom([shouldAnimate])**

滚动到内容底部。shouldAnimate参数为true|false，表示是否使用动画展示滚动过程

* **scrollTo(left,top[,shouldAnimate])**

滚动到指定位置。left和top分别表示要滚动到的x坐标和y坐标

* **scrollBy(left,top[,shouldAnimate])**

滚动指定偏移量。left和top分别表示要滚动的x偏移量和y偏移量

* **getScrollPosition()**

读取当前视图位置。返回值为一个JSON对象，具有left和top属性，分别表示x和y坐标

|  |
| --- |
| angular.module("myAPP",["ionic"])  .controller("firstCtrl",["$scope","$ionicScrollDelegate",function($scope,$ionicScollDelegate){  $scope.items = [];  for(var i=0;i<100;i++)  $scope.items.push(["item ",i+1].join(""));  $scope.gotop = function(){  $ionicScollDelegate.scrollTop(true);  }  $scope.gobottom = function(){  $ionicScollDelegate.scrollBottom(true);  }  }]); |