# UIScrollView

## contentOffset

默认CGPointZero，用来设置scrollView的滚动偏移量。

// 设置scrollView的滚动偏移量

scrollView.contentOffset = CGPointMake(0, 200);

## contentSize

默认CGSizeZero，用来设置scrollView的滚动范围。

// 设置scrollView的滚动范围

scrollView.contentSize = CGSizeMake( self.view.bounds.size.width, self.view.bounds.size.height \* 2);

## contentInset

默认 UIEdgeInsetsZero，用来设置scrollView的额外滚动区域。

// 设置scrollView的额外顶部滚动区域:(UIEdgeInsetsMake是逆时针设置，上左下右)

scrollView.contentInset = UIEdgeInsetsMake(100, 0, 0, 0);

1. bounces

默认为YES，用来设置scrollView的弹簧效果

// 取消scrollView的弹簧效果

scrollView.bounces = NO;

## 5.pagingEnabled

默认为NO，用来设置scrollView是否开启分页.

// 开启分页

scrollView.pagingEnabled = YES;

## 6.scrollEnabled

默认为YES，用来设置scrollView是否允许滚动.

// scrollView禁止滚动

scrollView.scrollEnabled = NO;

## 7.showsHorizontalScrollIndicator

默认为YES，用来设置scrollView是否显示水平滚动条.

// 隐藏水平滚动条

scrollView.showsHorizontalScrollIndicator = NO;

## 8.showsVerticalScrollIndicator

默认为YES，用来设置scrollView是否显示垂直滚动条.

// 隐藏垂直滚动条

scrollView.showsVerticalScrollIndicator = NO;

## 9.minimumZoomScale

默认1.0，用来设置scrollView最少缩小比例.

// 设置scrollView允许子视图的最大放大比例

scrollView.maximumZoomScale = 2;

## 10.maximumZoomScale

默认1.0，用来设置scrollView最大放大比例.

// 设置scrollView允许子视图的最小缩放比例

scrollView.minimumZoomScale = 0.8;

## 11. delegate

类要遵守UIScrollViewDelegate协议

scrollView.delegate = self;　声明scrollView的代理是我， 这个方法在UIScrollViewDelegate里所以我们要遵守UIScrollViewDelegate这个协议

### 11.1当scrollView的偏移量一改变就会调用这个方法，即滚动scrollView就会调用。

- (void)scrollViewDidScroll:(UIScrollView \*)scrollView;

### 11.2 当scrollView缩放时，就会调用这个方法

- (void)scrollViewDidZoom:(UIScrollView \*)scrollView NS\_AVAILABLE\_IOS(3\_2);

### 11.3 即将拖拽的时候调用.

- (void)scrollViewWillBeginDragging:(UIScrollView \*)scrollView;

### 11.4 即将停止拖拽的时候调用

- (void)scrollViewWillEndDragging:(UIScrollView \*)scrollView withVelocity:(CGPoint)velocity targetContentOffset:(inout CGPoint \*)targetContentOffset NS\_AVAILABLE\_IOS(5\_0);

### 11.5 停止拖拽的时候调用。

- (void)scrollViewDidEndDragging:(UIScrollView \*)scrollView willDecelerate:(BOOL)decelerate;

### 11.6 即将减速完成的时候调用。

- (void)scrollViewWillBeginDecelerating:(UIScrollView \*)scrollView;

### 11.7 减速完成的时候调用

- (void)scrollViewDidEndDecelerating:(UIScrollView \*)scrollView;

### 11.8 返回需要缩放哪个视图，这个视图必须是scrollView里的子视图。

- (UIView \*)viewForZoomingInScrollView:(UIScrollView \*)scrollView;

// 列如：

- (UIView \*)viewForZoomingInScrollView:(UIScrollView \*)scrollView

{

return scrollView.subviews[0];

}

### 11.9 scrollView即将开始缩放

- (void)scrollViewWillBeginZooming:(UIScrollView \*)scrollView withView:(UIView \*)view NS\_AVAILABLE\_IOS(3\_2);

### 11.10 scrollView完成缩放

- (void)scrollViewDidEndZooming:(UIScrollView \*)scrollView withView:(UIView \*)view atScale:(float)scale;

### 11.11没有开启分页的时候，代理方法调用顺序：

scrollViewWillBeginDragging –> scrollViewDidScroll -> scrollViewWillEndDragging -> scrollViewDidEndDragging

### 11.12 开启分页的时候，代理方法调用顺序

scrollViewWillBeginDragging -> scrollViewDidScroll -> scrollViewDidEndDragging ->scrollViewWillBeginDecelerating ->scrollViewDidScroll -> scrollViewDidEndDecelerating

## 12. directionalLockEnabled

指定控件是否只能在一个方向上滚动

## 13. decelerationRate

改变scrollerView的减速点位置

## 14. tracking

监控当前目标是否正在被跟踪

## 15. dragging

改变scrollerView的减速点位置

## 16. delaysContentTouches

控制视图是否延时调用开始滚动的方法

## 17. canCancelContentTouches

控制控件是否接触取消touch的事件

## 18. indicatorStyle

设定滚动条的样式

## 19. decelerating

监控当前目标是否正在减速