

# UICollectionView

## UICollectionView 三个显示元素

- 1、cell
- 2、页眉、页脚
- 3、装饰视图

## 如何实现UICollectionView

- 必须有布局对象
- 1、创建布局对象
  - 2、创建UICollectionView对象，设置数据源、代理

## 如何自定义布局类

```
UICollectionViewLayout
自定义布局类，必须继承UICollectionViewLayout，然后必须override 以下几个方法
//用于设置当前collectionView显示的大小，宽和高
1、- (CGSize)collectionViewContentSize
//在指定的区域（这个区域实际上就是collectionView的bounds），返回所有item的布局属性
2、- (NSArray *)layoutAttributesForElementsInRect:(CGRect)rect
//indexPath确定的具体的一个item，所以这个方法，主要目的就是返回指定的这个item的布局属性
3、- (UICollectionViewLayoutAttributes
*)layoutAttributesForItemAtIndexPath:(NSIndexPath *)indexPath

如果要实现页眉、页脚或者装饰视图的话， 还需要override其它的方法，详细说明参见官方手册！
```

## 如何完成插入、删除单元格

```

        if (tapRecognizer.state == UIGestureRecognizerStateEnded) {
            //需要确认一下， 当前的点击是发生在UICollectionView上， 还是发生在
            UICollectionViewCell上?
            // 1、确认一下当前手势发生的位置
            CGPoint currentPoint = [tapRecognizer
            locationInView:self.collectionView];
            // 2、根据手势发生位置，我们可以利用UICollectionView的对象方法生成一个
            NSIndexPath
            NSIndexPath *tappedIndexPath = [self.collectionView
            indexPathForItemAtPoint:currentPoint];
            // 3、判断点击的是CollectionView还是Cell? 由于NSIndexPath是与cell相关联， 所以
            如是NSIndexPath有值， 那么肯定是点击的cell， 否则点击不是cell
            // 点击到了item上
            if (tappedIndexPath != nil) {
                //实现单元格的删除动作
                self.cellCount -= 1;
                [self.collectionView performBatchUpdates:^(
                [self.collectionView
                deleteItemsAtIndexPaths:@[tappedIndexPath]]);
                } completion:NULL];

            }else //点击到的不是cell,而是UICollectionView
            {
                // 实现单元格的插入动作
                self.cellCount += 1;
                [self.collectionView performBatchUpdates:^(
                NSIndexPath *insertIndexPath = [NSIndexPath
                indexPathForItem:0 inSection:0];
                [self.collectionView
                insertItemsAtIndexPaths:@[insertIndexPath]];
                } completion:NULL];
            }
        }
    }
}

```

## 如何实现插入、删除单元格的动画效果

...

```

<!--确定添加元素开始的布局信息，主要用于插入item的开始动画的初始值，The collection view
uses this information as the starting point for any animations. -->

```

- (UICollectionViewLayoutAttributes)initialLayoutAttributesForAppearingItemAtIndexPath:(NSIndexPath)itemIndexPath { UICollectionViewLayoutAttributes \*attributes = [super initialLayoutAttributesForAppearingItemAtIndexPath:itemIndexPath]; if ([self.insertIndexPaths containsObject:itemIndexPath]) { if (!attributes) { attributes = [self layoutAttributesForItemAtIndexPath:itemIndexPath]; } attributes.alpha = 0.0; attributes.center = CGPointMake(self.center.x, self.center.y); } return attributes; }

- (UICollectionViewLayoutAttributes)*finalLayoutAttributesForDisappearingItemAtIndexPath:(NSIndexPath )*indexPath { UICollectionViewLayoutAttributes \*attributes = [super finalLayoutAttributesForDisappearingItemAtIndexPath:indexPath]; if ([self.deleteIndexPaths containsObject:indexPath]) { if (!attributes) { attributes = [self layoutAttributesForItemAtIndexPath:indexPath]; } attributes.alpha = 0.0; attributes.center = CGPointMake(self.center.x, self.center.y); attributes.transform3D = CATransform3DMakeScale(0.1, 0.1, 1.0); } return attributes; }

...