

# UITableView 的几种常见控件参数变更写法 (截止 -2016 年 10 月 26 日)

作者: [Leif160519](#)

原文链接: <https://ld246.com/article/1566554372880>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p><br>

UITableView 是 IOS 开发中几乎都会用到的,通常我们会对其中的一些参数进行自定义设定, 下面列常用的一些用法</p>

<p>首先做好准备工作, 写好带有 navigation 和 tableview 的页面并且在 cell 中填入数据,tableview 的类型先选择默认, cell 的类型也选择默认。 <br>

完整代码如下: </p>

<p>ViewController.m</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">#import "ViewController.h"
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">@interface ViewController ()&lt;UITableViewDataSource,UITableViewDelegate&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">@property (nonatomic,strong)UITableView * tableView;
</span></span><span class="highlight-line"><span class="highlight-cl">@end
</span></span><span class="highlight-line"><span class="highlight-cl">@implementation ViewController
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">-(void)viewDidLoad {
</span></span><span class="highlight-line"><span class="highlight-cl">    [super viewDidLoad];
</span></span><span class="highlight-line"><span class="highlight-cl">    // Do any additional setup after loading the view, typically from a nib.
</span></span><span class="highlight-line"><span class="highlight-cl">    _tableView = [[UITableView alloc] initWithFrame:self.view.bounds style:UITableViewStylePlain];
</span></span><span class="highlight-line"><span class="highlight-cl">    _tableView.dataSource = self;
</span></span><span class="highlight-line"><span class="highlight-cl">    _tableView.delegate = self;
</span></span><span class="highlight-line"><span class="highlight-cl">    [self.view addSubview:_tableView];
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">-(NSInteger)tableView:numberOfRowsInSection:(NSInteger)section
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl">    return 6;
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span><span class="highlight-line"><span class="highlight-cl">-(UITableViewCell)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl">    static NSString * ID = @"myCell";
</span></span><span class="highlight-line"><span class="highlight-cl">    UITableViewCell * cell = [tableView dequeueReusableCellWithIdentifier:ID];
</span></span><span class="highlight-line"><span class="highlight-cl">    if (cell == nil) {
</span></span><span class="highlight-line"><span class="highlight-cl">        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:ID];
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">    cell.textLabel.te
</span></span></pre>
```

```

t = [NSString stringWithFormat:@"第%d组,第%d行", (long)indexPath.section, (long)indexPath.
ow];
</span></span><span class="highlight-line"><span class="highlight-cl"> return cell;
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span><span class="highlight-line"><span class="highlight-cl">@end
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span></code></pre>
<p>实际效果: <br>
</p>
<h2 id="1-去掉多余的cell-">1.去掉多余的 cell: </h2>
<p>在 viewDidLoad 中添加如下代码: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">_mainTableView.tableFooterView = [[UIView alloc]init];
</span></span></code></pre>
<p>实际效果: <br>
</p>
<h2 id="2-去掉所有cell的分割线">2.去掉所有 cell 的分割线</h2>
<p>在 viewDidLoad 中添加如下代码: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">//设置分割线类型为无类型
</span></span><span class="highlight-line"><span class="highlight-cl">_mainTableView.s
paratorStyle = UITableViewCellStyleNone;
</span></span></code></pre>
<p>或: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">//设置分割线颜色为无色
</span></span><span class="highlight-line"><span class="highlight-cl">_mainTableView.s
paratorColor = [UIColor clearColor];
</span></span></code></pre>
<p>或在-(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSI
dexPath *)indexPath 中添加如下代码: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> //cell的边框颜色设置为白色, 边框宽度为2
</span></span><span class="highlight-line"><span class="highlight-cl"> cell.layer.border
olor = [[UIColor whiteColor]CGColor];
</span></span><span class="highlight-line"><span class="highlight-cl"> cell.layer.borde
Width = 2;
</span></span></code></pre>
<blockquote>
<p><strong>注意</strong>: *<strong>若要去掉部分 cell 分割线, 只需要判断一下 cell 的位置
可, 如: </strong></p>
</blockquote>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> if (indexPath.row == 2) {
</span></span><span class="highlight-line"><span class="highlight-cl"> cell.layer.bor
erColor = [[UIColor whiteColor]CGColor];
</span></span><span class="highlight-line"><span class="highlight-cl"> cell.layer.bor
erWidth = 2;
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span></code></pre>

```

</span></span></code></pre>

<p>去掉部分 cell 分割线实际效果: <br>

</p>

<p>去掉所有 cell 分割线实际效果: <br>

</p>

<h2 id="3-禁止选中整个tableview-">3.禁止选中整个 tableview: </h2>

<p>若想让添加在 tableview 上的子控件都不可选择, 可在 viewDidLoad 中添加如下代码: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">_mainTableView.allowsSelection = NO;
```

```
</span></span></code></pre>
```

<p>演示效果: <br>

</p>

<h2 id="4-禁止选中某一行cell-">4.禁止选中某一行 cell: </h2>

<p>若想单纯的禁止选中 cell 而 tableview 上其他子控件不受影响, 则在-(UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath 中添加如下代码: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">cell.selectionStyle = UITableViewCellSelectionStyleNone;
```

```
</span></span></code></pre>
```

<p>若想禁止具体某一行的 cell, 则只需定位到具体的 cell 行数即可: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">if (indexPath.row == 2) {
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">    cell.selectionStyle = UITableViewCellSelectionStyleNone;
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">    }
```

```
</span></span></code></pre>
```

<p>演示样例: <br>

</p>

<h2 id="5-cell分割线靠左-IOS10下全部有效--">5.cell 分割线靠左 (IOS10 下全部有效): </h2>

<p>默认的 cell 分割线是不靠左的, 想要靠左则需要添加如下代码: </p>

<p>方法一: <br>

在 viewDidLoad 中添加如下代码, 也是最简单的写法, 但是在之前的版本上可能无效: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">_mainTableView.separatorInset = UIEdgeInsetsMake(0, 0, 0, 0);
```

```
</span></span></code></pre>
```

<p>或者: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">_mainTableView.separatorInset = UIEdgeInsetsZero;
```

```
</span></span></code></pre>
```

<p>方法二: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">-(void)viewDidLoadSubviews
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">{
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">    if ([_mainTableView respondsToSelector:@selector(setSeparatorInset:)] {
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">        [_mainTableV
```

```

ew setSeparatorInset:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> if ([_mainTableV
ew respondsToSelector:@selector(setLayoutMargins:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [_mainTableV
ew setLayoutMargins:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">-(void)tableView:(
ITableView *)tableView willDisplayCell:(UITableViewCell *)cell forRowAtIndexPath:(NSIndexPat
*)indexPath
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl"> if ([cell respond
ToSelector:@selector(setSeparatorInset:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [cell setSepar
atorInset:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> if ([cell respond
ToSelector:@selector(setLayoutMargins:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [cell setLayout
Margins:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span></code></pre>

```

<p>方法三: <br>

直接在 viewDidLoad 中添加如下代码 (旧设备可能无效) : </p>

```

<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> if ([_mainTableView respondsToSelector:@selector(setSeparatorInset:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [_mainTableV
ew setSeparatorInset:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> if ([_mainTableV
ew respondsToSelector:@selector(setLayoutMargins:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [_mainTableV
ew setLayoutMargins:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span></code></pre>

```

<p>方法四: <br>

直接在-(UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSInd
xPath \*)indexPath 中添加如下代码(旧设备可能无效): </p>

```

<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> if ([cell respondsToSelector:@selector(setSeparatorInset:)] {
</span></span><span class="highlight-line"><span class="highlight-cl"> [cell setSepar
atorInset:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> if ([cell respond
ToSelector:@selector(setLayoutMargins:)] {

```

```

</span></span><span class="highlight-line"><span class="highlight-cl"> [cell setLayout
Margins:UIEdgeInsetsZero];
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span></code></pre>
<p>演示效果: <br>
</p>
<h2 id="6-点击cell时选中效果慢慢恢复-">6.点击 cell 时选中效果慢慢恢复: </h2>
<p>在 cell 点击事件方法中添加[tableView deselectRowAtIndexPath:indexPath animated:YES]
完整代码如下: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">-(void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)inde
Path
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl"> [tableView dese
ectRowAtIndexPath:indexPath animated:YES];
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span></code></pre>
<p>效果对比: <br>
 </p>
<h2 id="7-添加cell右侧指示器-">7.添加 cell 右侧指示器: </h2>
<p>在-(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSInd
xPath *)indexPath 中天下如下代码: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">cell.accessoryType = UITableViewCellAccessoryDisclosureIndicator;
</span></span></code></pre>
<p>实际效果: <br>
</p>
<p>#8.cell 的侧滑菜单 (自定义) </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">-(NSArray &lt;UITableViewRowAction *&gt;*)tableView:(UITableView *)tableView editActi
nsForRowAtIndex:(NSIndexPath *)indexPath
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl"> UITableViewRo
Action* deleteAction = [UITableViewRowAction rowActionWithStyle:UITableViewRowActionSt
yleDefault title:@"删除" handler:^(UITableViewRowAction *_Nonnull action,NSIndexPath * _No
null indexPath)
</span></span><span class="highlight-line"><span class="highlight-cl">
{
</span></span><span class="highlight-line"><span class="highlight-cl">
[_mainTableView deleteRowsAtIndexes:@[indexPath] withRowAnimation:UITableView
RowAnimationAutomatic];
</span></span><span class="highlight-line"><span class="highlight-cl">
_mainTableView.editing = NO;
</span></span><span class="highlight-line"><span class="highlight-cl">
}
</span></span></code></pre>

```

```

});
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> UITableViewRo
Action* editAction = [UITableViewRowAction rowActionWithStyle:UITableViewRowActionStyl
Normal title:@"编辑" handler:^(UITableViewRowAction *_Nonnull action,NSIndexPath *_Non
ull indexPath)
</span></span><span class="highlight-line"><span class="highlight-cl">
{
</span></span><span class="highlight-line"><span class="highlight-cl">
    [_mainTableView deleteRowsAtIndexPaths:@[indexPath] withRowAnimation:UITable
iewRowAnimationAutomatic];
</span></span><span class="highlight-line"><span class="highlight-cl">
    _mainTableView.editing = NO;
</span></span><span class="highlight-line"><span class="highlight-cl">
};
</span></span><span class="highlight-line"><span class="highlight-cl"> editAction.back
roundColor = [UIColor orangeColor];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> return @[delet
Action,editAction];
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span></code></pre>

```

<p>实际效果: <br>

</p>

<blockquote>

<p>\*\*注意: \*\*若想在不同的 cell 上显示不同的侧滑菜单, 则需要加上 row 的判断即可, 代码如下:

</p>

</blockquote>

```

<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> if (indexPath.row == 2) {
</span></span><span class="highlight-line"><span class="highlight-cl">         return @
deleteAction];
</span></span><span class="highlight-line"><span class="highlight-cl">     }
</span></span></code></pre>

```

<p>实例效果: <br>

</p>

<h2 id="9-系统自带cell侧滑删除按钮-">9.系统自带 cell 侧滑删除按钮: </h2>

<p>系统默认自带侧滑的菜单中只有删除功能, 代码如下: </p>

```

<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> -(NSString *)tableView:(UITableView *)tableView titleForDeleteConfirmationButtonForRo
AtIndex:(NSIndexPath *)indexPath
</span></span><span class="highlight-line"><span class="highlight-cl"> {
</span></span><span class="highlight-line"><span class="highlight-cl">     return @"删除";
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> //实现方法
</span></span><span class="highlight-line"><span class="highlight-cl"> -(void)tableView:(
ITableView *)tableView commitEditingStyle:(UITableViewCellEditingStyle)editingStyle forRowA
IndexPath:(NSIndexPath *)indexPath
</span></span><span class="highlight-line"><span class="highlight-cl"> {

```

```

</span></span><span class="highlight-line"><span class="highlight-cl"> NSLog(@"点击
删除按钮");
</span></span><span class="highlight-line"><span class="highlight-cl"> UILabel * v = [[
UILabel alloc] initWithFrame:CGRectMake(50, 350, self.view.bounds.size.width-100, 40)];
</span></span><span class="highlight-line"><span class="highlight-cl"> v.backgroundColor = [UIColor greenColor];
</span></span><span class="highlight-line"><span class="highlight-cl"> v.text = @"点
了删除按钮";
</span></span><span class="highlight-line"><span class="highlight-cl"> [self.view addS
bview:v];
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span></code></pre>
<p>实际效果<br>
</p>
<h2 id="10-长按cell自定义弹框-">10.长按 cell 自定义弹框:</h2>
<p>在-(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSInd
xPath *)indexPath 中给 cell 添加长按手势: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">UILongPressGestureRecognizer * longPress = [[UILongPressGestureRecognizer alloc]initW
thTarget:self action:@selector(longPress:)];
</span></span><span class="highlight-line"><span class="highlight-cl"> [cell addGestur
Recognizer:longPress];
</span></span></code></pre>
<p>实现方法: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">-(void)longPress:(UILongPressGestureRecognizer *)recognizer
</span></span><span class="highlight-line"><span class="highlight-cl">{
</span></span><span class="highlight-line"><span class="highlight-cl"> if (recognizer.st
te == UIGestureRecognizerStateBegan) {
</span></span><span class="highlight-line"><span class="highlight-cl"> //这里的UITablev
ewcell可以换成你自定义的cell
</span></span><span class="highlight-line"><span class="highlight-cl"> UITableView
ell * cell = (UITableViewCell *)recognizer.view;
</span></span><span class="highlight-line"><span class="highlight-cl"> [cell becomeF
rstResponder];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> // 获取row的
置
</span></span><span class="highlight-line"><span class="highlight-cl"> CGPoint poin
= [recognizer locationInView:_mainTableView];
</span></span><span class="highlight-line"><span class="highlight-cl"> NSIndexPath
path = [_mainTableView indexPathForRowAtPoint:point];
</span></span><span class="highlight-line"><span class="highlight-cl"> NSInteger r
w = [path row];
</span></span><span class="highlight-line"><span class="highlight-cl"> NSLog(@"
时row的值为%d",(long)row);
</span></span><span class="highlight-line"><span class="highlight-cl"> UIBarButtonItem
copy1 = [[UIBarButtonItem alloc] initWithTitle:@"复制" action:@selector(copy1:)];
</span></span><span class="highlight-line"><span class="highlight-cl"> UIBarButtonItem
transmit = [[UIBarButtonItem alloc] initWithTitle:@"转发" action:@selector(transmit:)];
</span></span><span class="highlight-line"><span class="highlight-cl"> UIBarButtonItem

```



```

widthdraw = [[UIMenuItem alloc] initWithTitle:@"撤回" action:@selector(widthdraw:)];
</span></span> <span class="highlight-line"> <span class="highlight-cl">
</span></span> <span class="highlight-line"> <span class="highlight-cl">      UIMenuCont
oller * menu = [UIMenuController sharedMenuController];
</span></span> <span class="highlight-line"> <span class="highlight-cl">      [menu setM
nultems:[NSArray arrayWithObjects:copy1,transmit,widthdraw, nil]];
</span></span> <span class="highlight-line"> <span class="highlight-cl">      [menu setTar
etRect:cell.frame inView:cell.superview];
</span></span> <span class="highlight-line"> <span class="highlight-cl">      [menu setM
nuVisible:YES animated:YES];
</span></span> <span class="highlight-line"> <span class="highlight-cl">    }
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> > //让长按成为第一
应,不写的话, 长按cell没反应
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >-(BOOL)canBeco
eFirstResponder
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >{
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >      return YES;
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >}
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> > //实现方法
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >-(void)copy1:(id)s
nder
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >{
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >      NSLog(@"点击
复制");
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >}
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >-(void)transmit:(id
sender
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >{
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >      NSLog(@"点击
转发");
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >}
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >-(void)widthdraw:(
d)sender
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >{
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >      NSLog(@"点击
撤回");
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >}
</span></span> <span class="highlight-line"> <span class="highlight-cl"> >}
</span></span> </code></pre>

```

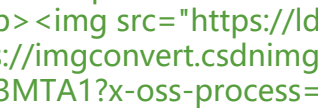
<p>效果演示<br>

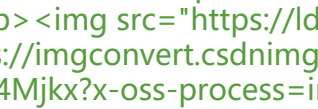
  </p>

<blockquote>

<p>\*\*注意: \*\*细心的朋友可能看到了我在写复制的时候, 定义的变量名不是 copy 而是 copy1, 为什么呢, 因为 IOS 本身自带 copy 的相关方法, 所以在写这些代码的时候, copy 的相关实现方法是可以动提示出来的, 因此我们常见的长按某项文字弹出的菜单, 只要有 copy 打头的, 基本都是用的系

自带的，如果我写的是 copy，那么英文的 copy 和中文的复制会一同出现





11.自定义 cell 的高度

cell 的默认高度为 44，我们也可以自定义 cell 的高度或者规定某一行的 cell 的固定高度，比如义所有 cell 的高度为 100:

方法一:

```
-(CGFloat)tableView:(UITableView *)tableView heightForRowAtIndexPath:(NSIndexPath *)indexPath {
    return 100;
}
```

方法二:

在 viewDidLoad 中添加如下代码:

```
_mainTableView.rowHeight = 100;
```

演示效果:



\*\*注意:\*\*以上两种方法仅针对改变所有 cell 高度，两个方法若同时写，则以方法二代码为准。

若要制定某一行 cell 的高度，则需要加判断即可:

```
-(CGFloat)tableView:(UITableView *)tableView heightForRowAtIndexPath:(NSIndexPath *)indexPath {
    switch (indexPath.row) {
        case 0:
            return 50;
            break;
        case 1:
            return 60;
            break;
        default:
            return 100;
            break;
    }
}
```

演示效果:



/imgconvert.csdnimg.cn/aHR0cDovL2ltZy5ibG9nLmNzZG4ubmV0LzlwMTYxMDI2MDk1MjU0DU0?x-oss-process=image/format,png" > </p>

<h2 id="12-设置cell中的数据-">12.设置 cell 中的数据: </h2>

<p>UITableViewCellStyle 有四种类型, 分别是: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">UITableViewCellStyleDefault,
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> // Simple cell with
text label and optional image view (behavior of UITableViewCell in iPhoneOS 2.x)
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leValue1,
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> // Left aligned lab
l on left and right aligned label on right with blue text (Used in Settings)
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leValue2,
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> // Right aligned l
abel on left with blue text and left aligned label on right (Used in Phone/Contacts)
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leSubtitle
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> // Left aligned lab
l on top and left aligned label on bottom with gray text (Used in iPod).
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> </code> </pre>
```

<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">分别解释为:

</span> </span> </code> </pre>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellStyleDefault-简单的单元格具有文本标签和可选图像视图
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leValue1-左边左对齐的标签和右边右对齐带有蓝色文本的标签
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leValue2-左边右对齐带有蓝色的标签和右边左对齐的标签
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> UITableViewCellSt
leSubtitle-顶部左对齐的标签和底部左对齐的带有灰色文本的标签
</span> </span> </code> </pre>
```

<p>为了对比明显我把代码整合在了一起, 完整代码如下: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> -(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> {
</span> </span> <span class="highlight-line"> <span class="highlight-cl">     static NSString
ID = @"myCell";
</span> </span> <span class="highlight-line"> <span class="highlight-cl">     UITableViewCell
* cell = [tableView dequeueReusableCellWithIdentifier:ID];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">     if (cell == nil) {
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl">     switch (indexPath
h.row) {
</span> </span> <span class="highlight-line"> <span class="highlight-cl">         case 0:
</span> </span> <span class="highlight-line"> <span class="highlight-cl">         {
</span> </span> <span class="highlight-line"> <span class="highlight-cl">             cell = [[UIT
bleViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:ID];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">             cell.backgr
undColor = [UIColor yellowColor];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">             cell.selecti
```

```

nStyle = UITableViewCellStyleDefault;
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">        break;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">        break;
</span></span><span class="highlight-line"><span class="highlight-cl">    case 1:
</span></span><span class="highlight-line"><span class="highlight-cl">    {
</span></span><span class="highlight-line"><span class="highlight-cl">        cell = [[UIT
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.backgr
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.selecti
nStyle = UITableViewCellStyleBlue;
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">        break;
</span></span><span class="highlight-line"><span class="highlight-cl">    case 2:
</span></span><span class="highlight-line"><span class="highlight-cl">    {
</span></span><span class="highlight-line"><span class="highlight-cl">        cell = [[UIT
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.backgr
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.selecti
nStyle = UITableViewCellStyleDefault;
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">        break;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">    case 3:
</span></span><span class="highlight-line"><span class="highlight-cl">    {
</span></span><span class="highlight-line"><span class="highlight-cl">        cell = [[UI
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.backgr
</span></span><span class="highlight-line"><span class="highlight-cl">        cell.selecti
nStyle = UITableViewCellStyleGray;
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">    cell.imageView.
image = [UIImage imageNamed:@"image.jpg"];
</span></span><span class="highlight-line"><span class="highlight-cl">    cell.detailTextL
bel.text = @"detailTextLabel";
</span></span><span class="highlight-line"><span class="highlight-cl">    cell.textLabel.te
t = @"textLabel";
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl">    return cell;
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span></code></pre>

```

**\*\*注意:\*\***系统自带的 cell 中的 UIImageView 是默认置顶和置底的，若想做成类似个人头像的 cell 布局必须自定义 cell 才可以！

演示效果如下：  


<h2 id="13-cell点击跳转页面-">13.cell 点击跳转页面: </h2>

<p>参见: <a href="https://ld246.com/forward?goto=http%3A%2F%2Fblog.csdn.net%2Fqq\_5139603%2Farticle%2Fdetails%2F52912399" target="\_blank" rel="nofollow ugc">http://blog.sdn.net/qq\_15139603/article/details/52912399</a> </p>