

# cell 自定义按钮删除崩溃

1. 简述: cellForRow 处理 cell 数据加载, 而事件处理一般来说会通过代理或者 block 回传到 Controller 进行具体的事件处理。不管通过哪种方式, 我们都需要知道用户点击了哪个 cell, 怎么来做具体的操作。这个看似简单的过程稍有不慎, 还是可能造成意想不到的后果的。比如接下来要介绍的删除事件
2. 错误代码逻辑:

```
- (void)tableView:(NotesIndexTableViewCell *)cell onClickDelete:(NSInteger)row {
    WeakObj(self);

    NotesDataModel *item = [self.tableModel getItemDataWithIndex:row];
    NSString *stockname = [NSString stringWithFormat:@"%@", item.stockname];
    NSString *stockcode = [NSString stringWithFormat:@"%@", item.stockcode];
    NSString *msg = [NSString stringWithFormat:@"%@", item.msg];
    if(weakSelf.tableView.lock == NO){
        weakSelf.tableView.lock = YES;

        UIAlertController *alert = [UIAlertController alertControllerWithTitle:@"提示" message:@"您确定删除该备忘吗?" preferredStyle:UIAlertControllerStyleAlert];
        UIAlertAction *cancel = [UIAlertAction actionWithTitle:@"取消" style:UIAlertActionStyleCancel handler:^(UIAlertAction * _Nonnull action) {
            weakSelf.tableView.lock = NO;
        }];
        UIAlertAction *confirm = [UIAlertAction actionWithTitle:@"确定" style:UIAlertActionStyleDefault handler:^(UIAlertAction * _Nonnull action) {

            [weakSelf.tableModel deleteNotesWithStockCode:stockcode withCompletion:^(BOOL isLastData, NSError *error) {
                if(error == nil){
                    NSInteger row = [weakSelf.tableModel.data indexOfObject:item];
                    [weakSelf.tableModel.data removeObject:item];
                    [weakSelf.tableView deleteRowsAtIndexPaths:@[[NSIndexPath indexPathForRow:row inSection:0]] withRowAnimation:UITableViewRowAnimationFade];
                    dispatch_after(dispatch_time(DISPATCH_TIME_NOW, (int64_t)(0.25 * NSEC_PER_SEC)), dispatch_get_main_queue(), ^{
                        weakSelf.tableView.lock = NO;
                    });
                }
            }else{
                NSString *msg = [NSString stringWithFormat:@"服务器繁忙(%ld)", error.code];
                Alert(@"提示", msg);
                weakSelf.tableView.lock = NO;
            }
        }];
    }
};
```

两个 cell, 删除 0 后删除 1, 读取 model 数据出现数组越界 bug。  
正确的获取 cell 位置的代码应该是通过 cell 去找  
正确代码逻辑:

```
row = [self.tableView indexPathForCell:cell].row;

NotesDataModel *item = [self.tableModel getItemDataWithIndex:row];
```

这种写法获取的 row 才是正确的 row

3. 分析  
其实很简单, tableView 的删除方法

## deleteRowsAtIndexPaths

会“刷新”列表, 并且带删除动画, 然而。。。这种方式并不会重新走 cellForRow, indexPath 无法得到更新, 导致上诉问题。如果是通过系统的编辑代理方法进

行删除的或者 `didselect` 方法里进行删除的就不会有这个问题，因为相应方法里返回的 `index` 是正确的。所以平时可能就没太注意。