**iOS开发,UITableViewCell基本操作（如:左滑删除按钮及编辑模式,全选,全不选等功能）**

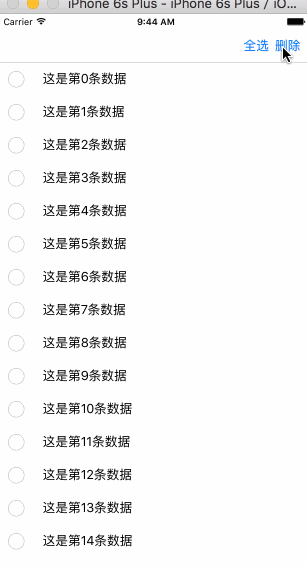
字数174 阅读137 评论3 喜欢6

* iOS开发,UITableViewCell基本操作,如:左滑删除按钮及编辑模式,全选,全不选等功能.
* [个人博客](https://nslog-yuhaitao.github.io/) : 个人博客主页
* [iOS频道(iOSPD)](http://www.jianshu.com/collection/d76ac79331c6): 这是我个人整理的一个技术专题, 这里的文章都是比较有技术含量(不断更新)!
* 微信公众号 :



*iOSPD微信公众号*

**代码区域**



*Display.gif*

#import "ViewController.h"

#define WIDTH [UIScreen mainScreen].bounds.size.width

#define HEIGHT [UIScreen mainScreen].bounds.size.height

@interface ViewController ()<UITableViewDelegate,UITableViewDataSource>

@property (nonatomic,strong) UITableView \* tableView;//列表

@property (nonatomic,strong) NSMutableArray \* dataArr;//数据源

@property (nonatomic,strong) NSMutableArray \* deleteArr;//存储删除的数据

@property (nonatomic,strong) UIBarButtonItem \* bar1;//编辑,删除按钮

@property (nonatomic,strong) UIBarButtonItem \* bar2;//全选,全不选按钮

@end

@implementation ViewController

- (void)viewDidLoad {

[super viewDidLoad];

self.view.backgroundColor = [UIColor whiteColor];

self.automaticallyAdjustsScrollViewInsets = NO;

[self createUI];

[self createDataArr];

[self createTableView];

}

#pragma mark --> createUI

- (void)createUI{

\_bar1 = [[UIBarButtonItem alloc]initWithTitle:@"编辑" style:UIBarButtonItemStylePlain target:self action:@selector(btnClick:)];

\_bar2 = [[UIBarButtonItem alloc]initWithTitle:@"全选" style:UIBarButtonItemStylePlain target:self action:@selector(AllBtnClick:)];

self.navigationItem.rightBarButtonItems = @[\_bar1];

}

- (void)btnClick:(UIBarButtonItem \*)item{

//点击编辑按钮

static BOOL flag = NO;

flag = !flag;

if (flag) {

// 编辑模式的时候可以多选

self.tableView.allowsMultipleSelectionDuringEditing = YES;

self.tableView.editing = YES;

\_bar1.title = @"删除";

self.navigationItem.rightBarButtonItems = @[\_bar1,\_bar2];

}else{

self.tableView.editing = NO;

\_bar1.title = @"编辑";

self.navigationItem.rightBarButtonItems = @[\_bar1];

[\_dataArr removeObjectsInArray:\_deleteArr];

[\_deleteArr removeAllObjects];

[\_tableView reloadData];

}

}

#pragma mark --> 全选按钮

- (void)AllBtnClick :(UIBarButtonItem \*)item{

static BOOL flag = NO;

flag = !flag;

if (flag) {

\_bar2.title = @"全不选";

// 获得选中的所有行

for (int i = 0; i < \_dataArr.count; i ++) {

NSIndexPath \*indexPath = [NSIndexPath indexPathForItem:i inSection:0];

[self.tableView selectRowAtIndexPath:indexPath animated:YES scrollPosition:UITableViewScrollPositionNone];

}

[self.deleteArr addObjectsFromArray:\_dataArr];

// NSLog(@"%@",\_deleteArr);

}else{

\_bar2.title = @"全选";

[self.tableView selectRowAtIndexPath:0 animated:YES scrollPosition:UITableViewScrollPositionNone];

[\_deleteArr removeAllObjects];

}

self.navigationItem.rightBarButtonItems = @[\_bar1,\_bar2];

}

#pragma mark --> 选中cell

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath{

[self.deleteArr addObject:\_dataArr[indexPath.row]];

NSLog(@"%@",\_deleteArr);

}

#pragma mark --> 取消选中cell

- (void)tableView:(UITableView \*)tableView didDeselectRowAtIndexPath:(NSIndexPath \*)indexPath{

NSLog(@"--------cell取消选中-------");

[self.deleteArr removeObject:\_dataArr[indexPath.row]];

}

#pragma mark --> createTableView

- (void)createTableView{

\_tableView = [[UITableView alloc]initWithFrame:CGRectMake(0, 64, WIDTH, HEIGHT - 64) style:UITableViewStylePlain];

\_tableView.delegate = self;

\_tableView.dataSource = self;

[self.view addSubview:\_tableView];

[\_tableView registerClass:[UITableViewCell class] forCellReuseIdentifier:@"CellID"];

//将文本放到最后一行

// NSIndexPath \* index = [NSIndexPath indexPathForRow:\_dataArr.count-1 inSection:0];

// [\_tableView scrollToRowAtIndexPath:index atScrollPosition:UITableViewScrollPositionBottom animated:YES];

}

#pragma mark --> createDataArr

- (void)createDataArr{

\_dataArr = [[NSMutableArray alloc]init];

for (int i = 0; i<40; i++) {

[\_dataArr addObject:[NSString stringWithFormat:@"这是第%d条数据",i]];

}

}

#pragma mark --> UITabelViewDelegate

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section{

return \_dataArr.count;

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath{

UITableViewCell \* cell = [tableView dequeueReusableCellWithIdentifier:@"CellID" forIndexPath:indexPath];

cell.textLabel.text = \_dataArr[indexPath.row];

return cell;

}

#pragma mark --> 修改cell上的文字

- (nullable NSString \*)tableView:(UITableView \*)tableView titleForDeleteConfirmationButtonForRowAtIndexPath:(NSIndexPath \*)indexPath{

return @"删除";

}

#pragma mark -- 左滑cell时出现什么按钮 -> Delegate方法

- (NSArray \*)tableView:(UITableView \*)tableView editActionsForRowAtIndexPath:(NSIndexPath \*)indexPath

{

UITableViewRowAction \*action0 = [UITableViewRowAction rowActionWithStyle:UITableViewRowActionStyleNormal title:@"置顶" handler:^(UITableViewRowAction \*action, NSIndexPath \*indexPath) {

NSLog(@"点击了置顶按钮");

// 收回左滑出现的按钮(退出编辑模式)

tableView.editing = NO;

}];

UITableViewRowAction \*action1 = [UITableViewRowAction rowActionWithStyle:UITableViewRowActionStyleDefault title:@"删除" handler:^(UITableViewRowAction \*action, NSIndexPath \*indexPath) {

NSLog(@"点击了删除按钮");

// 删除数据源

[\_dataArr removeObjectAtIndex:indexPath.row];

// 刷新数据

[tableView reloadData];

//或者cell删除

//[tableView deleteRowsAtIndexPaths:@[indexPath] withRowAnimation:UITableViewRowAnimationAutomatic];

}];

return @[action1, action0];

}

#pragma mark --> 懒加载

- (NSMutableArray \*)deleteArr{

if (\_deleteArr == nil) {

\_deleteArr = [[NSMutableArray alloc]init];

}

return \_deleteArr;

}

@end

Demo下载地址 : <https://github.com/NSLog-YuHaitao/UITableViewCell-Edit>

针对全选/取消还是存在问题,本人已将这块代码单独进行修改,请看:<http://www.jianshu.com/p/89a4e6ac72eb>

**声明**

* 所有文章出自 [Kenny Hito 的博客](https://nslog-yuhaitao.github.io/) !
* 未经本人允许不得转载, 转载请标明来源与作者, 谢谢合作!