

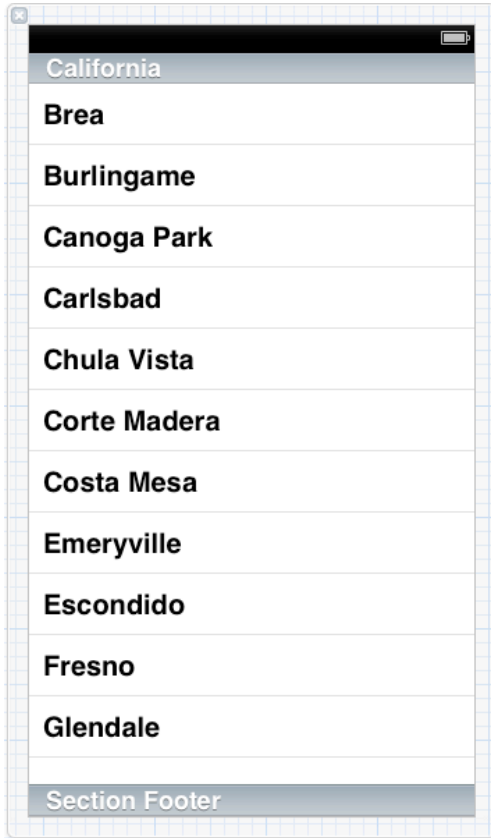
Gesture - Swipe on UITableViewCell

環境

- iOS 6.1
- Xcode 4.6.2

簡單的UITableView Sample

一開始從Single View Application開始，然後在ViewController的XIB放上一個UITableView。如下圖顯示這樣。



Listing. SSViewController.h

```
#import <UIKit/UIKit.h>

@interface SSViewController : UIViewController <UITableViewDelegate,
UITableViewDataSource> {
    IBOutlet UITableView* tv;
}
```

@end

實作UITableViewDelegate用來處理UITableView的事件，實作UITableViewDataSource來為UITableView提供資料。並且tv和XIB的UITableView bind起來。

Listing. SSViewController.m

```
-(void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view, typically from a nib.
    [self initView];
}

// initView
-(void)initView {
    tv.delegate = self;
    tv.dataSource = self;
}

// UITableViewDataSource methods
```

```

-(NSInteger)tableView:(UITableView *)tableView numberOfRowsInSectionSection:(NSInteger)section
{
    // 資料筆數
    return 3;
}

-(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:
(NSIndexPath *)indexPath {
    // 為了效能會reuse cell
    // reuse時間點是該cell重新需要顯示的時候
    // 所以一開始會看到null的log
    // 如果把table view往下拉讓第一個cell離開畫面
    // 再拉回來顯示第一個cell的時候就會看到reuse object log出來
    UITableViewCell* cell = [tv dequeueReusableCellWithIdentifier:@"cell"];
    NSLog( @"cell(%lu): %@", (unsigned long)[indexPath indexAtPosition:1], cell );
    if(nil == cell) {
        //cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
reuseIdentifier:@"cell"];
        // subtitle
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
reuseIdentifier:@"cell"];
    }

    // TODO: 怎麼讓這個work?
    //[cell setEditing:YES animated:YES];
    [[cell.textLabel] setText: [NSString stringWithFormat:@"Testing-%lu", (unsigned
long)[indexPath indexAtPosition:1]]];

    // subtitle
    [[cell.detailTextLabel] setText:@"SubTitle"];

    return cell;
}

// UITableViewDelegate methods
// cell 被點選的event
-(void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath
*)indexPath {
    [tableView deselectRowAtIndexPath:indexPath animated:YES];
}

```

接著就可以是跑看看會不會出現三個Testing的Cell。cellForRowAtIndexPath:裡的Log可以看到重複使用UITableViewCell的現象。之後也是要在這個method加上向右滑的手勢。

為UITableViewCell加上Swipe Gesture

在cellForRowAtIndexPath:可以為UITableViewCell加上手勢

Listing. 加上手勢的cellForRowAtIndexPath: 和處理手勢的handler

```

-(UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:
(NSIndexPath *)indexPath {
    // 為了效能會reuse cell
    // reuse時間點是該cell重新需要顯示的時候
    // 所以一開始會看到null的log
    // 如果把table view往下拉讓第一個cell離開畫面
    // 再拉回來顯示第一個cell的時候就會看到reuse object log出來
    UITableViewCell* cell = [tv dequeueReusableCellWithIdentifier:@"cell"];
    NSLog( @"cell(%lu): %@", (unsigned long)[indexPath indexAtPosition:1], cell );
    if(nil == cell) {
        //cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
reuseIdentifier:@"cell"];
        // subtitle
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
reuseIdentifier:@"cell"];
    }

    // TODO: 怎麼讓這個work?
    //[cell setEditing:YES animated:YES];
    [[cell.textLabel] setText: [NSString stringWithFormat:@"Testing-%lu", (unsigned
long)[indexPath indexAtPosition:1]]];

```

```

    // subtitle
    [[cell detailTextLabel] setText:@"SubTitle"];

    // 加上Gesture
    UISwipeGestureRecognizer* rightSwipe = [[UISwipeGestureRecognizer alloc]
initWithTarget:self action:@selector(swipeHandler:)];
    rightSwipe.direction = UISwipeGestureRecognizerDirectionRight;
    NSArray* swipes = [[NSArray alloc] initWithObjects:rightSwipe, nil];
    [cell setGestureRecognizers:swipes];
    [rightSwipe release];

    return cell;
}

// swip handler
-(void)swipeHandler:(UISwipeGestureRecognizer *)sender {
    NSLog( @"%@", sender );
    NSLog( @"%@", [sender view] );
    NSLog( @"%@", [[(UITableViewCell *)[sender view] titleLabel] text] );
    [[(UITableViewCell *)[sender view] detailTextLabel] setText:@"Left Swiped"];
}

```

手勢測試完畢～下一步就可以玩custom UITableViewCell了～