

# iOS关于在UITableView中，实现多个cell中不同的倒计时实现

在一个UITableView中，有多条数据，可能每一个cell对应的剩余时间不一样，所以，如何实现不同的cell中倒计时的实现？之前，考虑到需要单独为每一个cell中开启一个定时器，来监控对应cell的数据更新，但是很快发现这种方法行不通，因为不知道具体有多少条数据，这些数据都是动态从服务器获取的。所以，想到在请求最新的数据时，开启一个定时器，根据该定时器，分别对所有的需要进行倒计时显示的cell进行数据修改。最后，功能已经实现，核心代码如下：

//所有剩余时间数组

```
NSMutableArray *totalLastTime;
```

在网络请求到的所有数据中，根据需要将其中要进行倒计时显示的数据中的剩余时间单独保存出来，如果这里是所有的数据都需要倒计时，则只需要保存时间即可，如果是有部分数据才需要倒计时，则可以保存字典，两个键值对分别为其在UITableView的indexPath和剩余时间：num默认从0开始

```
NSDictionary *dic = @{@"indexPath":[NSString stringWithFormat:@"%i",num],@"lastTime": order.payLastTime};  
[totalLastTime addObject:dic];
```

开启定时器方法：

```
-(void)startTimer
```

```
{  
timer = [NSTimer scheduledTimerWithTimeInterval:1  
target:self selector:@selector(refreshLessTime) userInfo:@" repeats:YES];
```

如果不添加下面这条语句，在UITableView拖动的时候，会阻塞定时器的调用

```
[[NSRunLoop currentRunLoop] addTimer:timer  
forMode:UITrackingRunLoopMode];  
}
```

主要的定时器中的方法，在该方法中，遍历totalLastTime，取出其中保存的lasttime和indexpath，time用来显示，在显示完后自减，indexpath代表对应显示的位置，在一次循环后，将新的time和没有改变的indexpath从新替换totalLastTime 中对应位置的元素，以此保证每一秒执行时，显示time都是最

新的。

- (void)refreshLessTime

{

NSUInteger time;

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

time = [[[totalLastTime objectAtIndex:i] objectForKey:@'lastTime']integerValue];

NSIndexPath \*indexPath = [NSIndexPath indexPathForItem:0 inSection:

[[[totalLastTime objectAtIndex:i] objectForKey:@'indexPath'] integerValue]];

WLServiceOrderTableViewCell \*cell = (WLServiceOrderTableViewCell \*)

[\_tableView cellForRowAtIndexPath:indexPath:indexPath];

cell.remainingTimeLabel.text = [NSString stringWithFormat:@'剩余支付时间:

%@',[self lessSecondToDay:--time]]];

NSDictionary \*dic = @{@'indexPath':

[NSString stringWithFormat:@'%i',indexPath.section],@'lastTime':

[NSString stringWithFormat:@'%i',time]};

[totalLastTime replaceObjectAtIndex:i withObject:dic];

}

}

- (NSString \*)lessSecondToDay:(NSUInteger)seconds

{

NSUInteger day = (NSUInteger)seconds/(24\*3600);

NSUInteger hour = (NSUInteger)(seconds%(24\*3600))/3600;

NSUInteger min = (NSUInteger)(seconds%(3600))/60;

NSUInteger second = (NSUInteger)(seconds%60);

NSString \*time = [NSString stringWithFormat:@'%lu日 %lu小时%lu分钟%lu秒',

(unsigned long)day,(unsigned long)hour,(unsigned long)min,(unsigned

long)second];

return time;

}