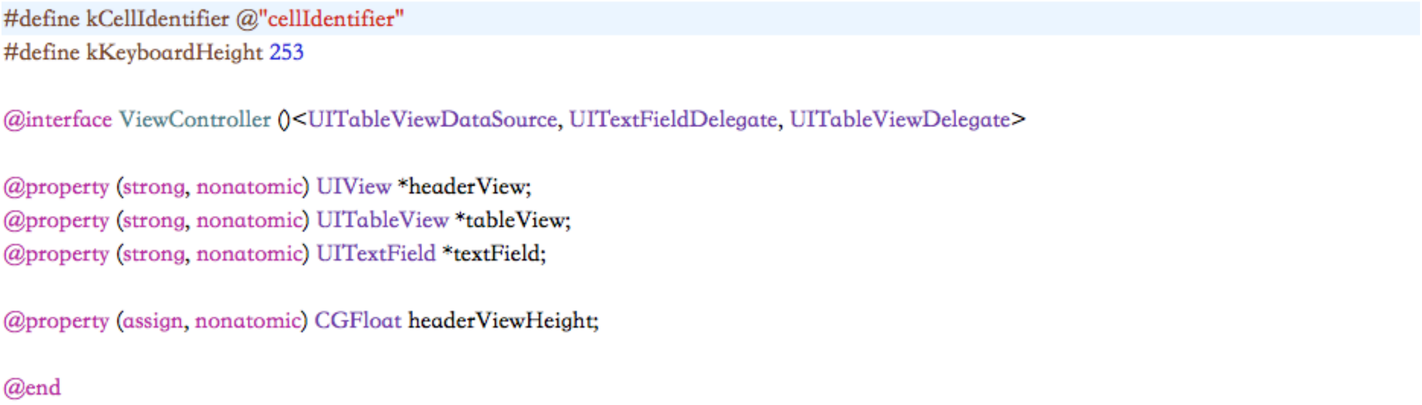
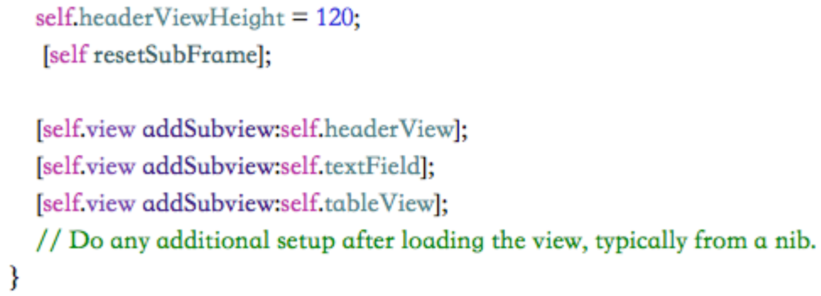
使用 frame 怎么实现 autolayout

1. 简述：界面绘制有两种方式，一种是用 frame 进行布局，另一种是使用 autolayout。后者好处是如果改变某个控件的一个约束，其它控件根据约束会自动做相应调整，相比下 frame 进行布局的则没有这么方便，要改好多控件的 frame。当然，直接使用 frame 布局也有一些独有的好处，比如：比较好做控件不同尺寸屏幕的适配。这里要探究的是，如果采用 frame 进行布局，怎么实现自动布局，当一个控件的位置或大小改变了，其它控件会自动调整新的 frame。
2. 解决：autolayout 主要过程是，做好所有控件的相对约束，当其中一个约束改变的时候，系统会根据约束重新所有想对控件的 frame。那么使用 frame 布局的，能不能也根据这种思想，先做好相对位置，然后仿系统做出重设控件 frame 所应有的操作过程？答案是肯定的，具体例子如下：

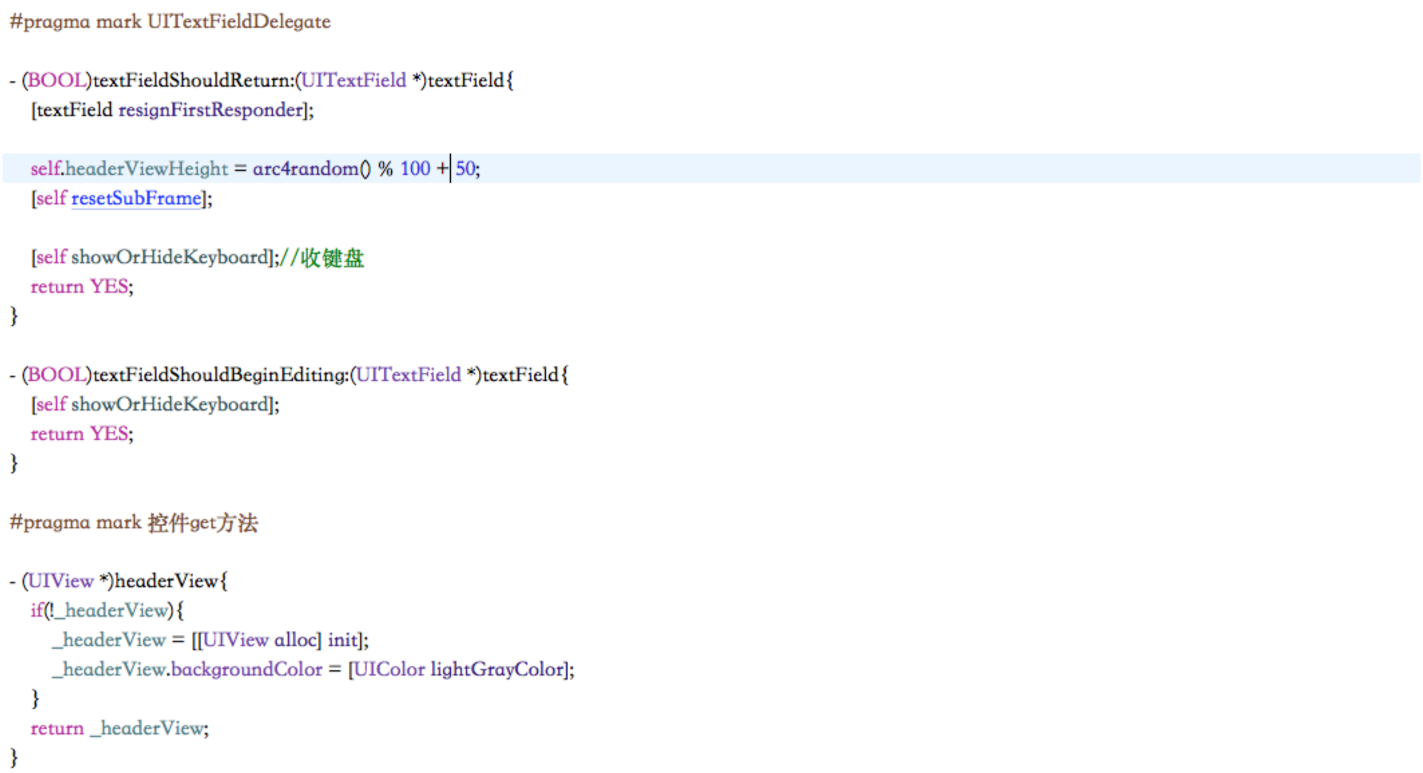


```
#define kCellIdentifier @"cellIdentifier"
#define kKeyboardHeight 253

@interface ViewController ()<UITableViewDataSource, UITextFieldDelegate, UITableViewDelegate>

@property (strong, nonatomic) UIView *headerView;
@property (strong, nonatomic) UITableView *tableView;
@property (strong, nonatomic) UITextField *textField;

@property (assign, nonatomic) CGFloat headerViewHeight;

@end
```

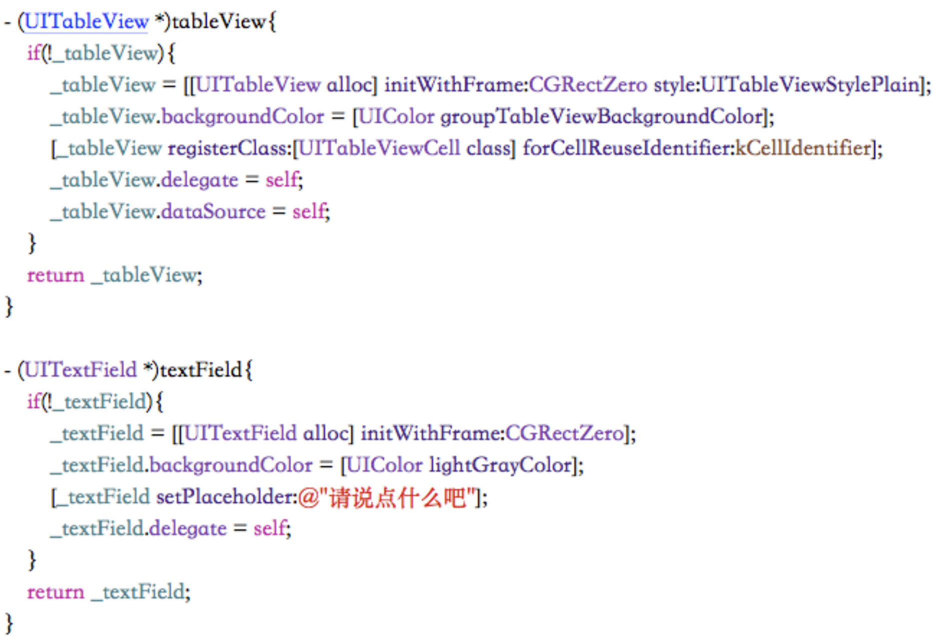


```
    self.headerViewHeight = 120;
     [self resetSubFrame];

    [self.view addSubview:self.headerView];
    [self.view addSubview:self.textField];
    [self.view addSubview:self.tableView];
    // Do any additional setup after loading the view, typically from a nib.
}
```

```
#pragma mark private methods

- (void)resetSubFrame{
    [UIView animateWithDuration:0.25 animations:^{
        self.headerView.frame = CGRectMake(0, 0, self.view.frame.size.width, self.headerViewHeight);
        self.tableView.frame = CGRectMake(0, CGRectGetMaxY(self.headerView.frame), self.view.frame.size.width, self.view.frame.size.height - CGRectGetMaxY(self.headerView.frame) -44);
        self.textField.frame = CGRectMake(0, self.view.frame.size.height - 44, self.view.frame.size.width, 44);
    }];
}

- (void)showOrHideKeyboard{
    CGRect viewRect = self.view.frame;
    CGFloat duration = 0.12;
    if(viewRect.origin.y < 0){//键盘显示，收键盘
        viewRect.origin.y = 0;
    }else{
        duration = 0.45;
        viewRect.origin.y  = -kKeyboardHeight;
    }

    [UIView animateWithDuration:duration animations:^{
            self.view.frame = viewRect;
    }];
}

#pragma mark UITableViewDataSource

- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView{
    return 1;
}

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section{
    return 20;
}

- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath{
    UITableViewCell *cell;
    cell = [tableView dequeueReusableCellWithIdentifier:kCellIdentifier forIndexPath:indexPath];
    if(cell == nil){
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:kCellIdentifier];
    }
    [cell.textLabel setText:@"fdsaf"];
    return cell;
}

#pragma mark UITableViewDelegate

- (void)scrollViewWillBeginDragging:(UIScrollView *)scrollView{
    if(self.view.frame.origin.y < 0){//收键盘
        [self showOrHideKeyboard];
        [self.textField resignFirstResponder];
    }
}

- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath{
    [tableView deselectRowAtIndexPath:indexPath animated:YES];
}
```



```
#pragma mark UITextFieldDelegate

- (BOOL)textFieldShouldReturn:(UITextField *)textField{
    [textField resignFirstResponder];

    self.headerViewHeight = arc4random() % 100 +50;
    [self resetSubFrame];

    [self showOrHideKeyboard];//收键盘
    return YES;
}

- (BOOL)textFieldShouldBeginEditing:(UITextField *)textField{
    [self showOrHideKeyboard];
    return YES;
}

#pragma mark 控件get方法

- (UIView *)headerView{
    if(!_headerView){
        _headerView = [[UIView alloc] init];
        _headerView.backgroundColor = [UIColor lightGrayColor];
    }
    return _headerView;
}
```



```
- (UITableView *)tableView{
    if(!_tableView){
        _tableView = [[UITableView alloc] initWithFrame:CGRectZero style:UITableViewStylePlain];
        _tableView.backgroundColor = [UIColor groupTableViewBackgroundColor];
        [_tableView registerClass:[UITableViewCell class] forCellReuseIdentifier:kCellIdentifier];
        _tableView.delegate = self;
        _tableView.dataSource = self;
    }
    return _tableView;
}

- (UITextField *)textField{
    if(!_textField){
        _textField = [[UITextField alloc] initWithFrame:CGRectZero];
        _textField.backgroundColor = [UIColor lightGrayColor];
        [_textField setPlaceholder:@"请说点什么吧"];
        _textField.delegate = self;
    }
    return _textField;
}
```

关键方法为

```
-  (void)resetSubFrame;
```

控件初始化时 frame 可以设为 0，并调用一次 resetSubFrame，resetSubFrame 里面写的是所有控件相对关系，对于某些可变的值要写成可变属性，在属性改变后重新调用一下 resetSubFrame 方法即可实现自动布局。