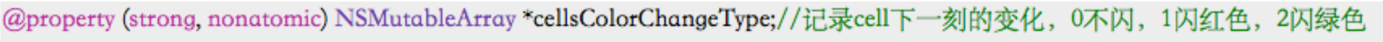
cell 渐变色背景闪烁动画及重用问题

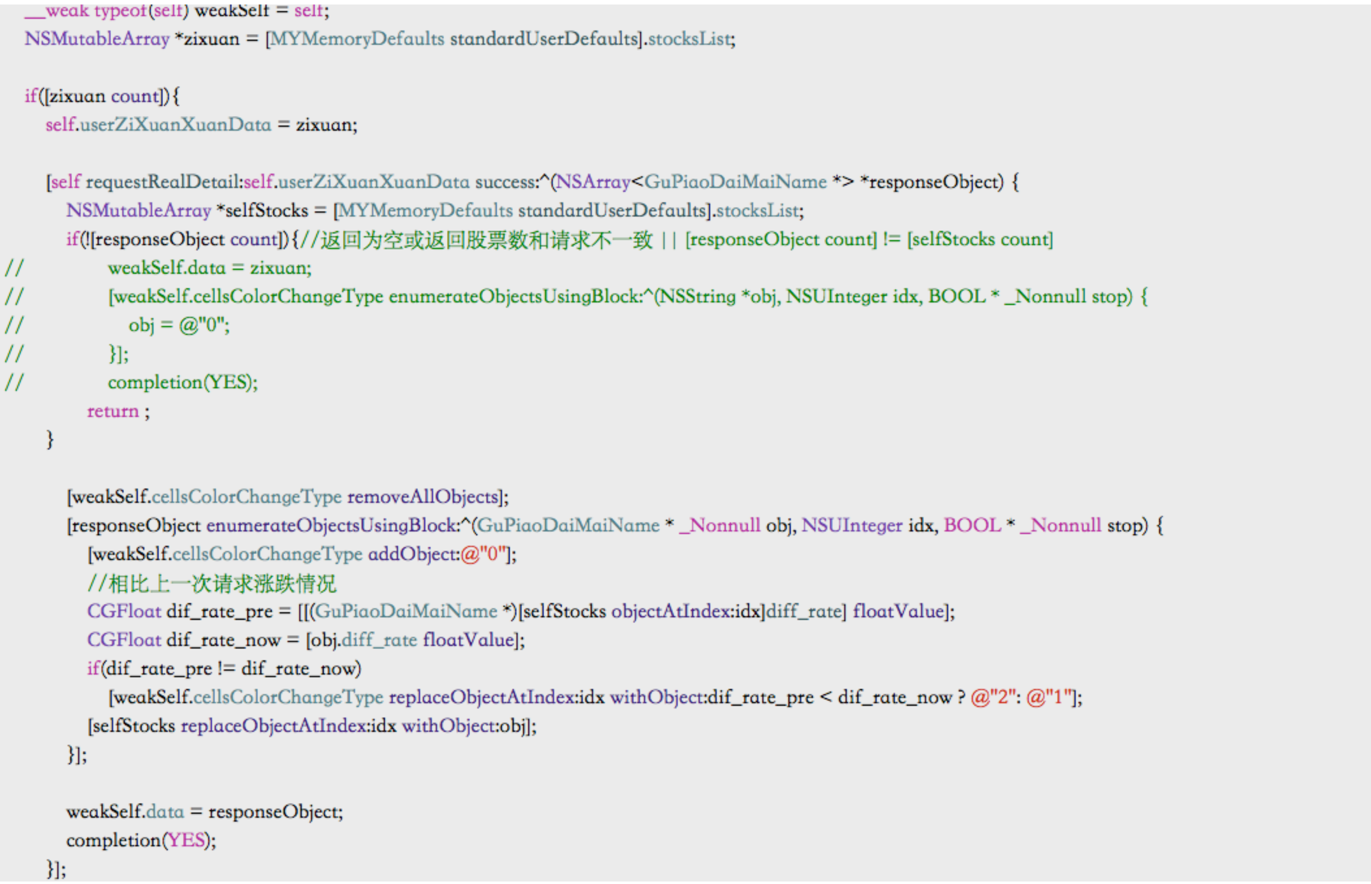
1. 简述：股票行情资讯项目中，在用户自选股股票行情发生改变，比如相比上一次数据股票价上涨，则整个 cell 白红渐变色闪烁一下提醒用户，这只股票这个时刻上涨了，跌了则用绿色提醒用户。除了动画功能实现我们还需考虑 cell 动画重用的问题。

2. 实现：

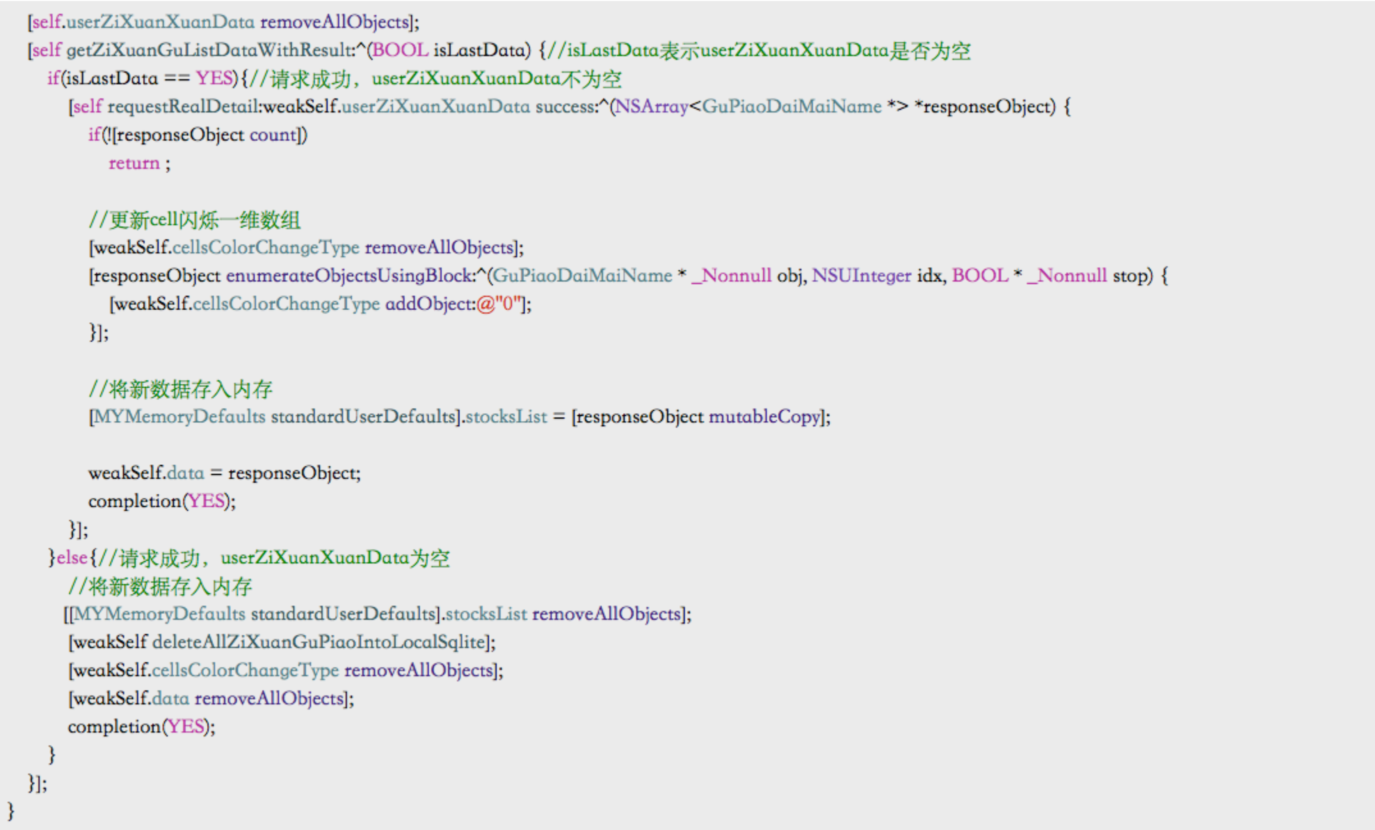
   (1) 数据大小判断



```
@property (strong, nonatomic) NSMutableArray *cellsColorChangeType;//记录cell下一刻的变化，0不闪，1闪红色，2闪绿色
```

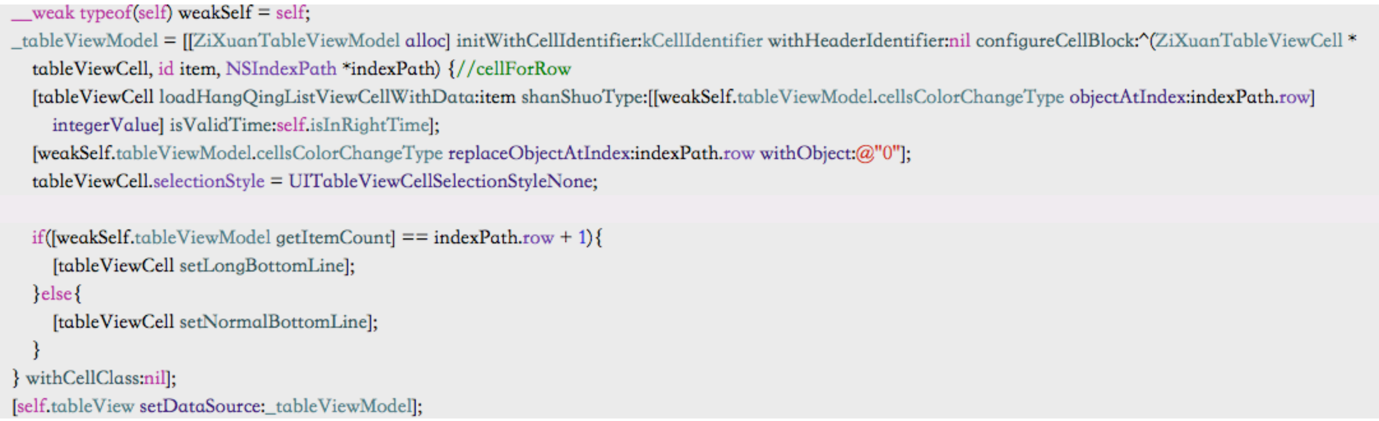


```
  __weak typeof(self) weakSelf = self;
  NSMutableArray *zixuan = [MYMemoryDefaults standardUserDefaults].stocksList;

  if([zixuan count]){
     self.userZiXuanXuanData = zixuan;

     [self requestRealDetail:self.userZiXuanXuanData success:^(NSArray<GuPiaoDaiMaiName *> *responseObject) {
        NSMutableArray *selfStocks = [MYMemoryDefaults standardUserDefaults].stocksList;
        if(![responseObject count]){//返回为空或返回股票数和请求不一致 || [responseObject count] != [selfStocks count]
//            weakSelf.data = zixuan;
//            [weakSelf.cellsColorChangeType enumerateObjectsUsingBlock:^(NSString *obj, NSUInteger idx, BOOL * _Nonnull stop) {
//               obj = @"0";
//            }];
//            completion(YES);
           return ;
     }

        [weakSelf.cellsColorChangeType removeAllObjects];
        [responseObject enumerateObjectsUsingBlock:^(GuPiaoDaiMaiName * _Nonnull obj, NSUInteger idx, BOOL * _Nonnull stop) {
           [weakSelf.cellsColorChangeType addObject:@"0"];
           //相比上一次请求涨跌情况
           CGFloat dif_rate_pre = [[(GuPiaoDaiMaiName *)[selfStocks objectAtIndex:idx]diff_rate] floatValue];
           CGFloat dif_rate_now = [obj.diff_rate floatValue];
           if(dif_rate_pre != dif_rate_now)
              [weakSelf.cellsColorChangeType replaceObjectAtIndex:idx withObject:dif_rate_pre < dif_rate_now ? @"2": @"1"];
           [selfStocks replaceObjectAtIndex:idx withObject:obj];
        }];

        weakSelf.data = responseObject;
        completion(YES);
     }];
```

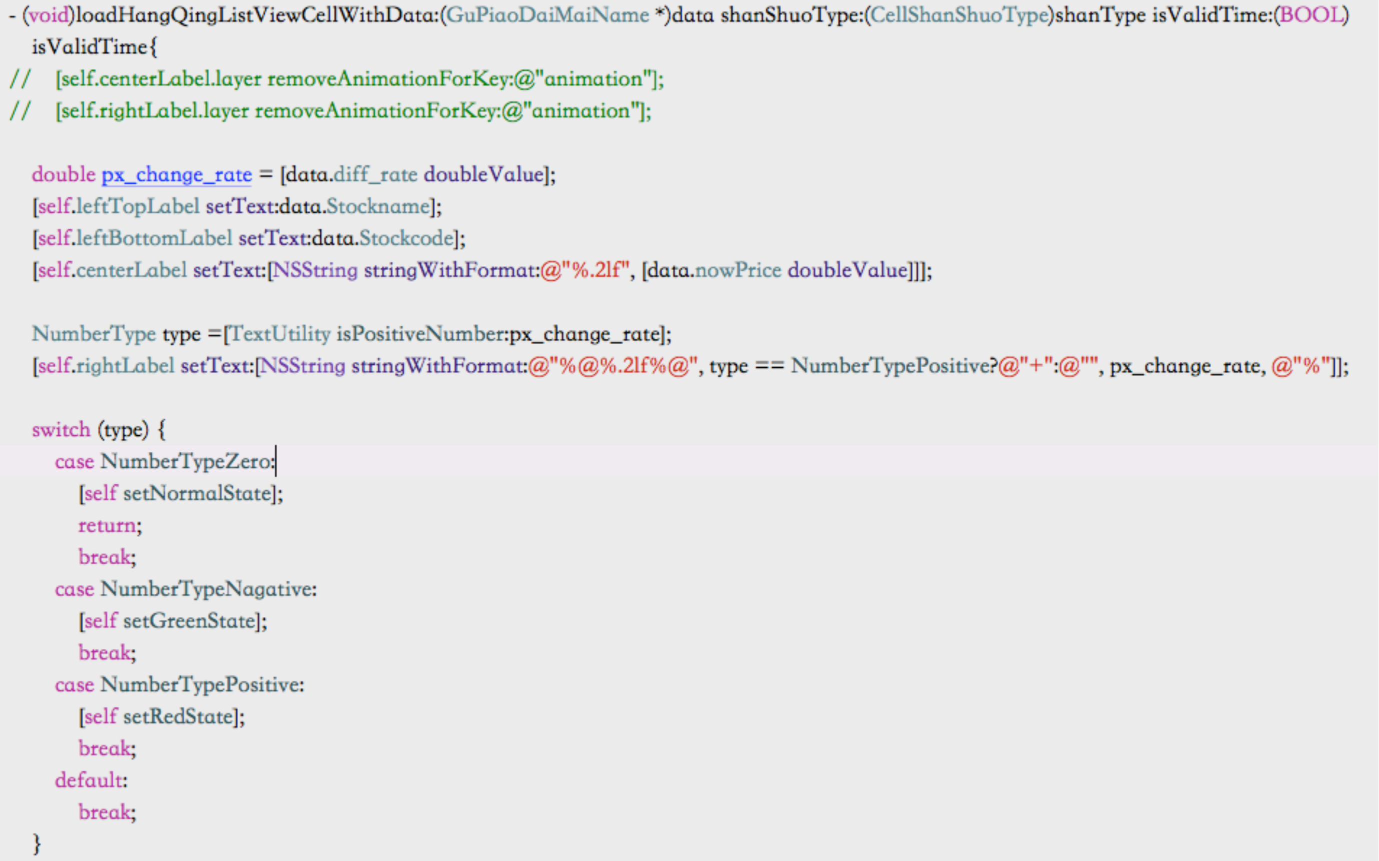


```
    [self.userZiXuanXuanData removeAllObjects];
    [self getZiXuanGuListDataWithResult:^(BOOL isLastData) {//isLastData表示userZiXuanXuanData是否为空
        if(isLastData == YES){//请求成功，userZiXuanXuanData不为空
            [self requestRealDetail:weakSelf.userZiXuanXuanData success:^(NSArray<GuPiaoDaiMaiName *> *responseObject) {
                if(![responseObject count])
                    return ;

                //更新cell闪烁一维数组
                [weakSelf.cellsColorChangeType removeAllObjects];
                [responseObject enumerateObjectsUsingBlock:^(GuPiaoDaiMaiName * _Nonnull obj, NSUInteger idx, BOOL * _Nonnull stop) {
                    [weakSelf.cellsColorChangeType addObject:@"0"];
                }];

                //将新数据存入内存
                [MYMemoryDefaults standardUserDefaults].stocksList = [responseObject mutableCopy];

                weakSelf.data = responseObject;
                completion(YES);
            }];
        }else{//请求成功，userZiXuanXuanData为空
            //将新数据存入内存
           [[MYMemoryDefaults standardUserDefaults].stocksList removeAllObjects];
            [weakSelf deleteAllZiXuanGuPiaoIntoLocalSqlite];
            [weakSelf.cellsColorChangeType removeAllObjects];
            [weakSelf.data removeAllObjects];
            completion(YES);
        }
    }];
}
```

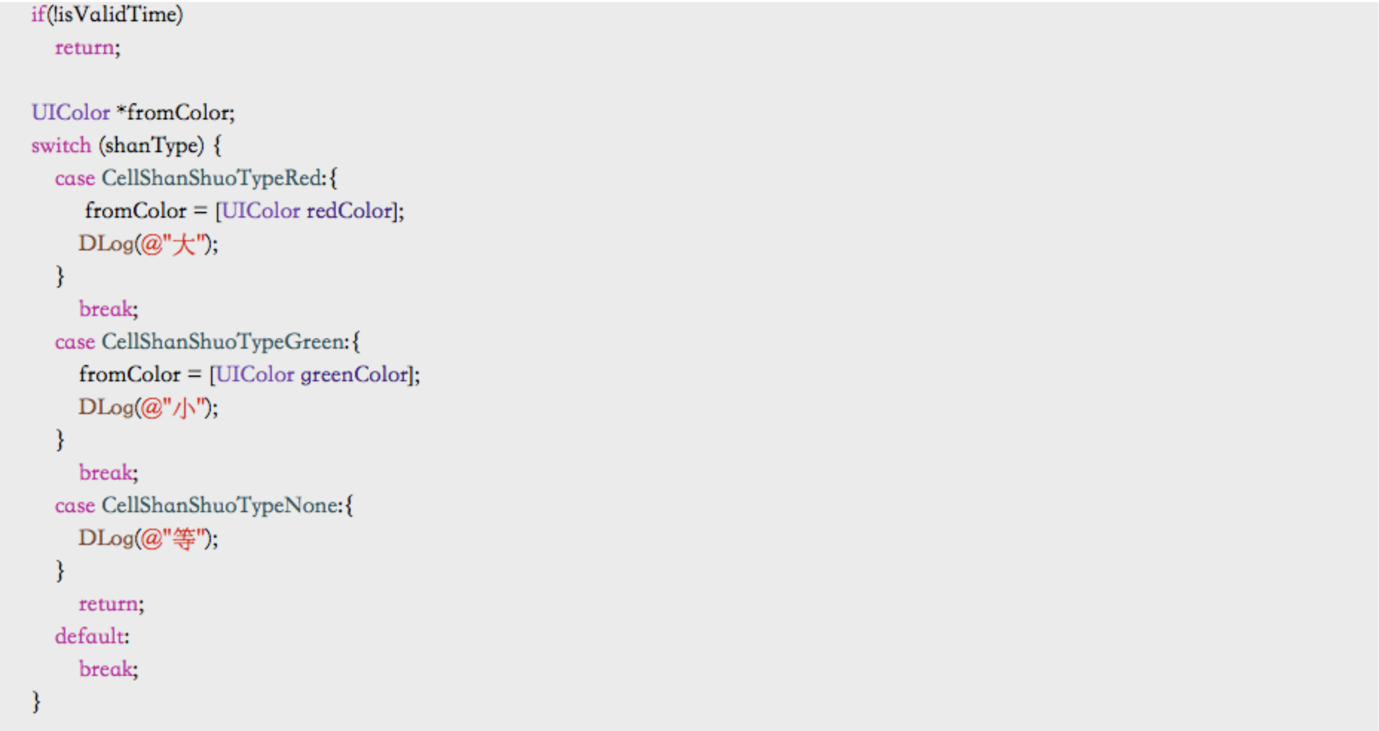
(2) cell 动画实现



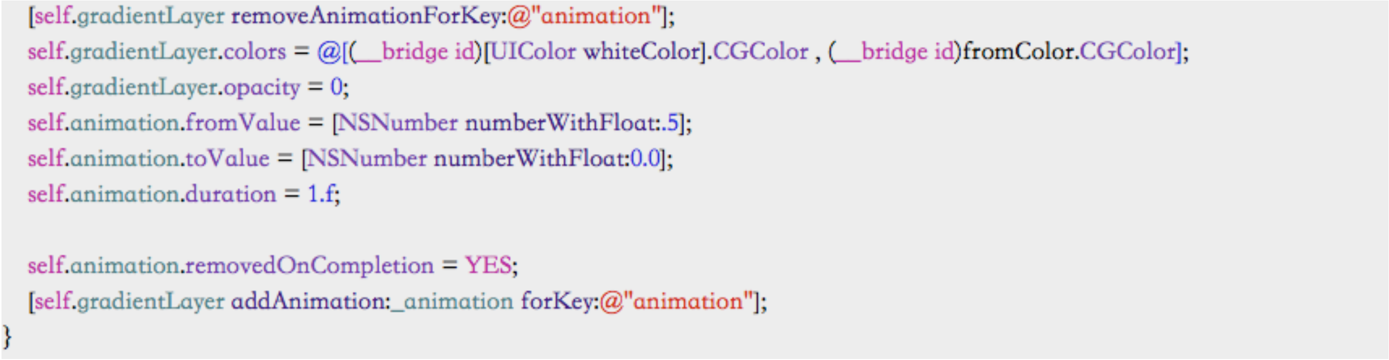
```
__weak typeof(self) weakSelf = self;
_tableViewModel = [[ZiXuanTableViewModel alloc] initWithCellIdentifier:kCellIdentifier withHeaderIdentifier:nil configureCellBlock:^(ZiXuanTableViewCell *
    tableViewCell, id item, NSIndexPath *indexPath) {//cellForRow
    [tableViewCell loadHangQingListViewCellWithData:item shanShuoType:[[weakSelf.tableViewModel.cellsColorChangeType objectAtIndex:indexPath.row]
        integerValue] isValidTime:self.isInRightTime];
    [weakSelf.tableViewModel.cellsColorChangeType replaceObjectAtIndex:indexPath.row withObject:@"0"];
    tableViewCell.selectionStyle = UITableViewCellSelectionStyleNone;

    if([weakSelf.tableViewModel getItemCount] == indexPath.row + 1){
        [tableViewCell setLongBottomLine];
    }else{
        [tableViewCell setNormalBottomLine];
    }
} withCellClass:nil];
[self.tableView setDataSource:_tableViewModel];
```



```
- (void)loadHangQingListViewCellWithData:(GuPiaoDaiMaiName *)data shanShuoType:(CellShanShuoType)shanType isValidTime:(BOOL)
  isValidTime{
//    [self.centerLabel.layer removeAnimationForKey:@"animation"];
//    [self.rightLabel.layer removeAnimationForKey:@"animation"];

   double px_change_rate = [data.diff_rate doubleValue];
   [self.leftTopLabel setText:data.Stockname];
   [self.leftBottomLabel setText:data.Stockcode];
   [self.centerLabel setText:[NSString stringWithFormat:@"%.2lf", [data.nowPrice doubleValue]]];

   NumberType type =[TextUtility isPositiveNumber:px_change_rate];
   [self.rightLabel setText:[NSString stringWithFormat:@"%@%.2lf%@", type == NumberTypePositive?@"+":@"", px_change_rate, @"%"]];

   switch (type) {
      case NumberTypeZero:
         [self setNormalState];
         return;
         break;
      case NumberTypeNagative:
         [self setGreenState];
         break;
      case NumberTypePositive:
         [self setRedState];
         break;
      default:
         break;
   }
```

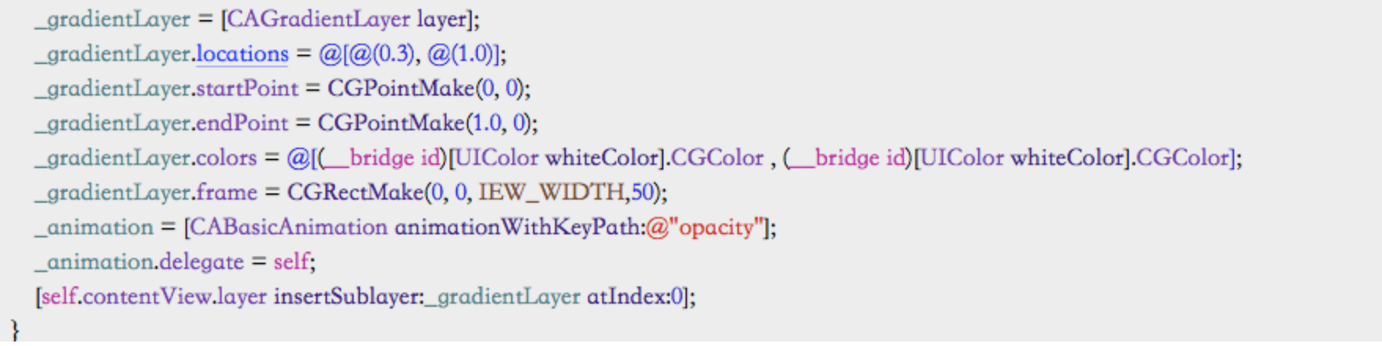


```
   if(!isValidTime)
      return;

   UIColor *fromColor;
   switch (shanType) {
      case CellShanShuoTypeRed:{
          fromColor = [UIColor redColor];
         DLog(@"大");
      }
         break;
      case CellShanShuoTypeGreen:{
         fromColor = [UIColor greenColor];
         DLog(@"小");
      }
         break;
      case CellShanShuoTypeNone:{
         DLog(@"等");
      }
         return;
      default:
         break;
   }
```

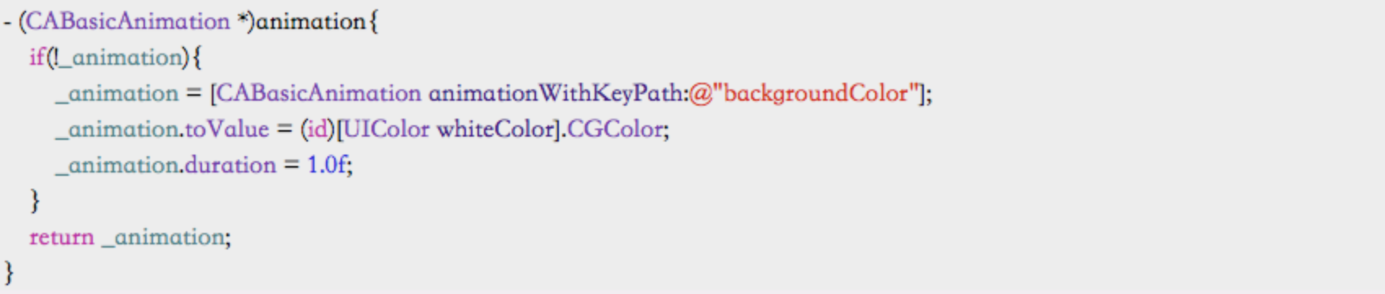


```
  [self.gradientLayer removeAnimationForKey:@"animation"];
  self.gradientLayer.colors = @[(__bridge id)[UIColor whiteColor].CGColor , (__bridge id)fromColor.CGColor];
  self.gradientLayer.opacity = 0;
  self.animation.fromValue = [NSNumber numberWithFloat:.5];
  self.animation.toValue = [NSNumber numberWithFloat:0.0];
  self.animation.duration = 1.f;

  self.animation.removedOnCompletion = YES;
  [self.gradientLayer addAnimation:_animation forKey:@"animation"];
}
```

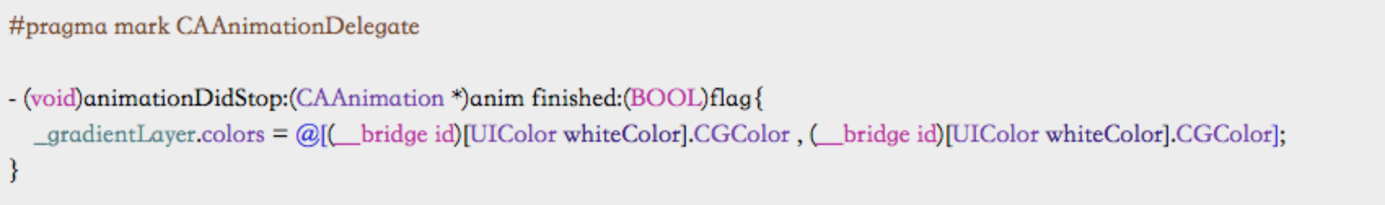
渐变色图层及 animation 动画



```
    _gradientLayer = [CAGradientLayer layer];
    _gradientLayer.locations = @[@(0.3), @(1.0)];
    _gradientLayer.startPoint = CGPointMake(0, 0);
    _gradientLayer.endPoint = CGPointMake(1.0, 0);
    _gradientLayer.colors = @[(__bridge id)[UIColor whiteColor].CGColor , (__bridge id)[UIColor whiteColor].CGColor];
    _gradientLayer.frame = CGRectMake(0, 0, IEW_WIDTH,50);
    _animation = [CABasicAnimation animationWithKeyPath:@"opacity"];
    _animation.delegate = self;
    [self.contentView.layer insertSublayer:_gradientLayer atIndex:0];
}
```



```
- (CABasicAnimation *)animation{
    if(!_animation){
        _animation = [CABasicAnimation animationWithKeyPath:@"backgroundColor"];
        _animation.toValue = (id)[UIColor whiteColor].CGColor;
        _animation.duration = 1.0f;
    }
    return _animation;
}
```



```
#pragma mark CAAnimationDelegate

- (void)animationDidStop:(CAAnimation *)anim finished:(BOOL)flag{
    _gradientLayer.colors = @[(__bridge id)[UIColor whiteColor].CGColor , (__bridge id)[UIColor whiteColor].CGColor];
}
```

(3) 关于动画重用问题，只需，在 cell 动画执行代码前前把前面的动画关闭即可