HomeIcon 菜单：

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions

{

    // 用来处理3D Touch触发事件

    if  ([UIDevice currentDevice].systemVersion.floatValue >= 9.0) {

        [self addShortcutItems];

    }

}

// 用来处理3D Touch触发事件

- (void)application:(UIApplication \*)application performActionForShortcutItem:(UIApplicationShortcutItem \*)shortcutItem completionHandler:(void(^)(BOOL succeeded))completionHandler

{

    //判断先前我们设置的唯一标识

    if([shortcutItem.localizedTitle isEqualToString:@"标签1"]){

        NSArray \*arr = @[@"hello 3D Touch"];

        UIActivityViewController \*vc = [[UIActivityViewController alloc]initWithActivityItems:arr applicationActivities:nil];

        //设置当前的VC 为rootVC

        [self.window.rootViewController presentViewController:vc animated:YES completion:^{

        }];

    }

    if ([shortcutItem.localizedTitle isEqual: @"标签2"]) {

        // 点击标签2时，显示提示框

        UIAlertController \*alertC = [UIAlertController alertControllerWithTitle:@"OPPS!" message:@"啦啦啦" preferredStyle:UIAlertControllerStyleAlert];

        UIAlertAction \*alert = [UIAlertAction actionWithTitle:@"确定" style:UIAlertActionStyleDefault handler:^(UIAlertAction \* \_Nonnull action) {

        }];

        [alertC addAction:alert];

        [self.window.rootViewController presentViewController:alertC animated:YES completion:^{

        }];

        return;

    }

}

- (void)addShortcutItems

{

    UIApplicationShortcutItem \* item = [[UIApplicationShortcutItem alloc]initWithType:@"two" localizedTitle:@"标签1" localizedSubtitle:@"222" icon:[UIApplicationShortcutIcon iconWithType:UIApplicationShortcutIconTypeLove] userInfo:nil];

    // 设置自定义标签图片

    UIApplicationShortcutItem \* itemTwo = [[UIApplicationShortcutItem alloc]initWithType:@"two" localizedTitle:@"标签2" localizedSubtitle:@"333" icon:[UIApplicationShortcutIcon iconWithTemplateImageName:@"account\_icon\_clock"] userInfo:nil];

    UIApplicationShortcutItem \* itemThird = [[UIApplicationShortcutItem alloc]initWithType:@"two" localizedTitle:@"标签3" localizedSubtitle:@"444" icon:[UIApplicationShortcutIcon iconWithType:UIApplicationShortcutIconTypeSearch] userInfo:nil];

    [UIApplication sharedApplication].shortcutItems = @[item, itemTwo, itemThird];

}

POP：

- (NSArray \*)previewActionItems {

    UIPreviewAction \*action1 = [UIPreviewAction actionWithTitle:@"Action 1" style:UIPreviewActionStyleDefault handler:^(UIPreviewAction \* \_Nonnull action, UIViewController \* \_Nonnull previewViewController) {

        // 点击此选项触发

        UIAlertController \*alertC = [UIAlertController alertControllerWithTitle:@"你查看的是:" message:[NSString stringWithFormat:@"图片"] preferredStyle:UIAlertControllerStyleActionSheet];

        UIAlertAction \*alert = [UIAlertAction actionWithTitle:@"噢噢" style:UIAlertActionStyleDefault handler:^(UIAlertAction \* \_Nonnull action) {

        }];

        [alertC addAction:alert];

        // 因为预览视图与根视图不在一个视图层级上，所以需要通过根视图去推出这个

        [[UIApplication sharedApplication].keyWindow.rootViewController presentViewController:alertC animated:YES completion:^{

        }];

    }];

    // 可以添加多个选项

    NSArray \*arr = @[action1];

    return arr;

}

PEEK:

//LDPMHallViewControllerr

-(void)viewDidLoad {

    if (self.traitCollection.forceTouchCapability == UIForceTouchCapabilityAvailable) {

        // 注册预览视图的代理和来源视图

        [self registerForPreviewingWithDelegate:(id)self sourceView:self.view];

    }

}

}

// If you return nil, a preview presentation will not be performed

- (nullable UIViewController \*)previewingContext:(id <UIViewControllerPreviewing>)previewingContext viewControllerForLocation:(CGPoint)location NS\_AVAILABLE\_IOS(9\_0)

{

    if (CGRectContainsPoint(self.marketInfoCell.frame, location)) {

        //previewingContext.sourceRect = CGRectMake(0, 0, 100,100);//最开始轻按时除了这个框外，都变模糊

        NPMProductViewController \*productViewController = [NPMProductViewController new];

        for(LDPMHallMarketInfoView \*marketInfoView in [self recursionViewListFromView:self.marketInfoCell ofSomeKind:[LDPMHallMarketInfoView class]]) {

            CGRect convertedRect = [marketInfoView convertRect:marketInfoView.bounds toView:self.view];

            if (CGRectContainsPoint(convertedRect, location)) {

                NPMProduct \*product = [NPMProduct productWithGoodsId:marketInfoView.marketInfo.goodsId goodsName:marketInfoView.marketInfo.goodsName partnerId:marketInfoView.marketInfo.partnerId];

                for (NPMProduct \*obj in self.hotGoodsList) {

                    if ([obj.partnerId isEqualToString:product.partnerId] && [obj.goodsId isEqualToString:product.goodsId]) {

                        product.enableTrade = obj.enableTrade;

                    }

                }

                productViewController.product = product;

                productViewController.marketInfo = marketInfoView.marketInfo;

                break;

            }

        }

        //productViewController.preferredContentSize = CGSizeMake(0, 300);//预览框的大小

        return productViewController;

    }

    return nil;

}

- (NSArray \*)recursionViewListFromView:(UIView \*)aView ofSomeKind:(Class)aClass

{

    NSMutableArray \*mutableArray = [NSMutableArray array];

    for (UIView \*subView in aView.subviews) {

        if ([subView isKindOfClass:aClass]) {

            [mutableArray addObject:subView];

        } else {

            NSArray \*recursionArray = [self recursionViewListFromView:subView ofSomeKind:aClass];

            [mutableArray addObjectsFromArray:recursionArray];

        }

    }

    return [mutableArray copy];

}

- (void)previewingContext:(id <UIViewControllerPreviewing>)previewingContext commitViewController:(UIViewController \*)viewControllerToCommit NS\_AVAILABLE\_IOS(9\_0)

{

    [self.navigationController pushViewController:viewControllerToCommit animated:YES];

}

如果是tableview

// If you return nil, a preview presentation will not be performed

- (nullable UIViewController \*)previewingContext:(id <UIViewControllerPreviewing>)previewingContext viewControllerForLocation:(CGPoint)location NS\_AVAILABLE\_IOS(9\_0)

{

    if (CGRectContainsPoint(self.marketInfoView.frame, location)) {

        //previewingContext.sourceRect = CGRectMake(0, 0, 100,100);//最开始轻按时除了这个框外，都变模糊

        NPMProductViewController \*productViewController = [NPMProductViewController new];

        for(LDPMHallMarketInfoView \*marketInfoView in [self recursionViewListFromView:self.marketInfoView ofSomeKind:[LDPMHallMarketInfoView class]]) {

            CGRect convertedRect = [marketInfoView convertRect:marketInfoView.bounds toView:self.view];

            if (CGRectContainsPoint(convertedRect, location)) {

                NPMProduct \*product = [NPMProduct productWithGoodsId:marketInfoView.marketInfo.goodsId goodsName:marketInfoView.marketInfo.goodsName partnerId:marketInfoView.marketInfo.partnerId];

                for (NPMProduct \*obj in self.hotGoodsList) {

                    if ([obj.partnerId isEqualToString:product.partnerId] && [obj.goodsId isEqualToString:product.goodsId]) {

                        product.enableTrade = obj.enableTrade;

                    }

                }

                productViewController.product = product;

                productViewController.marketInfo = marketInfoView.marketInfo;

                break;

            }

        }

        //productViewController.preferredContentSize = CGSizeMake(0, 300);//预览框的大小

        return productViewController;

    } else {

        return nil;

    }

}