- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString \*)reuseIdentifier {

self = [super initWithStyle:style reuseIdentifier:reuseIdentifier];

if (self) {

[self.contentView addSubview:self.bg];

[self.bg addSubview:self.smallBg];

[self.smallBg addSubview:self.timeLab];

[self.smallBg addSubview:self.nameLab];

[self.smallBg addSubview:self.againPostBtn];

}

return self;

}

- (UIView \*)bg {

if (!\_bg) {

bg = [[UIView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, 55)];

bg.backgroundColor = MyCellColor;

}

return \_bg;

}

- (UIView \*)smallBg {

if (!\_smallBg) {

smallBg = [[UIView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, 50)];

smallBg.backgroundColor = [UIColor whiteColor];

}

return \_smallBg;

}

- (UILabel \*)timeLab {

if (!\_timeLab) {

timeLab = [[UILabel alloc] initWithFrame:CGRectMake(10, 0, 200, 20)];

timeLab.text = @"编辑时间:2016-7012- 15:00";

timeLab.font = [UIFont systemFontOfSize:14];

}

return \_timeLab;

}

- (UILabel \*)nameLab {

if (!\_nameLab) {

nameLab = [[UILabel alloc] initWithFrame:CGRectMake(10, 21, 200, 30)];

nameLab.text = @"产品:硫酸钾";

nameLab.font = [UIFont boldSystemFontOfSize:16];

}

return \_nameLab;

}

- (UIButton \*)againPostBtn {

if (!\_againPostBtn) {

againPostBtn = [UIButton buttonWithType:UIButtonTypeCustom];

againPostBtn.frame = CGRectMake(kDeviceWidth - 75, 12.5, 60, 25);

[\_againPostBtn setTitle:@"重新发布" forState:UIControlStateNormal];

[\_againPostBtn.titleLabel setFont:[UIFont systemFontOfSize:14]];

[\_againPostBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

againPostBtn.layer.borderWidth = 1.0f;

againPostBtn.layer.borderColor = kOrangeColor.CGColor;

againPostBtn.layer.masksToBounds = YES;

againPostBtn.layer.cornerRadius = 5.0;

againPostBtn.backgroundColor = [UIColor whiteColor];

}

return \_againPostBtn;

}

#import "BJMyDraftVC.h"

#import "MyDraftViewCell.h"

#import "PublishProducInfotVC.h"

@interface BJMyDraftVC ()<UITableViewDelegate, UITableViewDataSource>

@property (nonatomic, strong) UITableView \*myDraftTab;

@property (nonatomic, strong) MyDraftViewCell \*myDraftCell;

@property(nonatomic, assign) int page;

@property(nonatomic, assign) NSUInteger allCount;

@property (nonatomic, strong) NSMutableArray \*dataArr;

@property (nonatomic, strong)UILabel \*sorryLabel;

@end

@implementation BJMyDraftVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.title = @"草稿箱";

\_dataArr = [@[] mutableCopy];

\_myDraftTab = [[UITableView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight - 64) style:UITableViewStylePlain];

\_myDraftTab.delegate = self;

\_myDraftTab.dataSource = self;

[self.view addSubview:\_myDraftTab];

[self setExtraCellLineHidden:\_myDraftTab];

[\_myDraftTab setSeparatorInset:UIEdgeInsetsMake(0, 0, 0, 0)];

\_sorryLabel = [[UILabel alloc]initWithFrame:CGRectMake(0, self.view.frame.size.height/2.0-50, kDeviceWidth, 30)];

\_sorryLabel.text = @"暂无附近业务";

\_sorryLabel.textAlignment = NSTextAlignmentCenter;

\_sorryLabel.textColor = [UIColor blackColor];

\_sorryLabel.font = [UIFont systemFontOfSize:17];

\_page = 0;

//请求数据

[self getDataWithNum:\_page];

\_myDraftTab.header = [MJRefreshNormalHeader headerWithRefreshingBlock:^{

[self pullDownRereshing];

}];

\_myDraftTab.footer = [MJRefreshBackNormalFooter footerWithRefreshingBlock:^{

[self getmoreData];

}];

}

//下拉刷新数据

- (void)pullDownRereshing {

\_page = 0;

[\_dataArr removeAllObjects];

[self getDataWithNum:\_page];

}

//上拉 加载 更多

- (void) getmoreData {

if (\_dataArr.count < \_allCount) {

page++;

[self getDataWithNum:\_page];

} else {

NSLog(@"没有数据了");

[self endRefreshingFromHeaderOrFooter];

return;

}

}

#pragma mark----- UITableViewDataSource

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath {

return 55;

}

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section {

//return self.myAddressArr.count;

return \_dataArr.count;

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

static NSString \*identifer = @"myDraftCell";

\_myDraftCell = [tableView dequeueReusableCellWithIdentifier:identifer];

if (!\_myDraftCell) {

myDraftCell = [[MyDraftViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:identifer];

}

\_myDraftCell.timeLab.text = [NSString stringWithFormat:@"%@ %@", \_dataArr[indexPath.row][@"ctime"], \_dataArr[indexPath.row][@"cstrtime"]];

\_myDraftCell.nameLab.text = [NSString stringWithFormat:@"%@", \_dataArr[indexPath.row][@"product"]];

[\_myDraftCell.againPostBtn addTarget:self action:@selector(postAction:event:) forControlEvents:UIControlEventTouchUpInside];

return \_myDraftCell;

}

#pragma mark------Action

- (void)postAction:(id)sender1 event:(id)event {

NSLog(@"重新发布！！");

NSSet \*touches =[event allTouches];

UITouch \*touch =[touches anyObject];

CGPoint currentTouchPosition = [touch locationInView:\_myDraftTab];

NSIndexPath \* indexPath = [\_myDraftTab indexPathForRowAtPoint:currentTouchPosition];

NSString \*orderID = [NSString stringWithFormat:@"%@", \_dataArr[indexPath.row][@"orderid"]];

PublishProducInfotVC \*publishVC = [[PublishProducInfotVC alloc] init];

publishVC.orderID = orderID;

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

}

//侧滑删除

- (NSArray<UITableViewRowAction \*> \*)tableView:(UITableView \*)tableView editActionsForRowAtIndexPath:(NSIndexPath \*)indexPath {

UITableViewRowAction \*deleteRowAction = [UITableViewRowAction rowActionWithStyle:UITableViewRowActionStyleDestructive title:@"删除" handler:^(UITableViewRowAction \* \_Nonnull action, NSIndexPath \* \_Nonnull indexPath) {

NSString \*orderID = \_dataArr[indexPath.row][@"orderid"];

NSLog(@"序号:%@", orderID);

[self deleteMyDraftDataWith:orderID];

if (\_dataArr.count > indexPath.row) {

//1.删除数据源

[\_dataArr removeObjectAtIndex:indexPath.row];

//2.删除UI界面

[\_myDraftTab deleteRowsAtIndexPaths:@[indexPath] withRowAnimation:UITableViewRowAnimationAutomatic];

}

}];

return @[deleteRowAction];

}

//删除数据

- (void)deleteMyDraftDataWith:(NSString \*)orderID {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSMutableDictionary \*param = [[NSMutableDictionary alloc] init];

[param setValue:orderID forKey:@"orderid"];

[BJHttpTool Post:Kdeletefb parameters:param success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*roorDict = [[NSMutableDictionary alloc] init];

roorDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

if ([[roorDict objectForKey:@"errcode"] intValue] == 0) {

[self getDataWithNum:\_page];

}

}

[\_myDraftTab reloadData];

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

[lbxManager showHudViewLabelText:@"删除失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

[lbxManager showHUDAction:1 showView:self.view];

}

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath {

}

#pragma mark---- 数据处理

- (void)getDataWithNum:(int) num {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSLog(@"num = %d", num);

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSString \*username = [NSString stringWithFormat:@"%@", dict[@"username"]];

NSString \*numStr = [NSString stringWithFormat:@"%d", num];

NSDictionary \*parma = [NSDictionary dictionaryWithObjectsAndKeys:username,@"username",@"待发布", @"status",numStr,@"page", nil];

NSLog(@"什么鬼 = %@", parma);

[BJHttpTool post:Knewapi3GetFbList3 parameters:parma success:^(id responseObject) {

NSError \*error;

NSMutableDictionary \*rootDict = [[NSMutableDictionary alloc] init];

rootDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

if ([rootDict[@"errcode"] integerValue] == 0) {

[lbxManager showHudViewLabelText:@"获取数据成功" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

NSLog(@"rootDict = %@", rootDict);

allCount = [[NSString stringWithFormat:@"%@", [NSString stringWithFormat:@"%@", rootDict[@"count"]]] integerValue];

if (\_allCount != 0) {

[\_sorryLabel removeFromSuperview];

} else {

[self.view addSubview:\_sorryLabel];

}

for (NSDictionary \*dic in rootDict[@"list"]) {

[\_dataArr addObject:dic];

}

} else {

NSLog(@"数据请求错误!");

}

[\_myDraftTab reloadData];

[self endRefreshingFromHeaderOrFooter];

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

[lbxManager showHudViewLabelText:@"获取数据失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

[self endRefreshingFromHeaderOrFooter];

}];

[lbxManager showHUDAction:1 showView:self.view];

}

// 结束刷新

- (void)endRefreshingFromHeaderOrFooter

{

NSLog(@"结束刷新了啊刷新了啊");

[\_myDraftTab.footer endRefreshing];

[\_myDraftTab.header endRefreshing];

}

- (void)setExtraCellLineHidden:(UITableView \*)tableView {

UIView \*view = [UIView new];

view.backgroundColor = [UIColor clearColor];

[tableView setTableFooterView:view];

}

-(void)viewDidLayoutSubviews {

if ([\_myDraftTab respondsToSelector:@selector(setSeparatorInset:)]) {

[\_myDraftTab setSeparatorInset:UIEdgeInsetsZero];

}

if ([\_myDraftTab respondsToSelector:@selector(setLayoutMargins:)]){

[\_myDraftTab setLayoutMargins:UIEdgeInsetsZero];

}

}

-(void)tableView:(UITableView \*)tableView willDisplayCell:(UITableViewCell \*)cell forRowAtIndexPath:(NSIndexPath \*)indexPat{

if ([cell respondsToSelector:@selector(setLayoutMargins:)]) {

[cell setLayoutMargins:UIEdgeInsetsZero];

}

if ([cell respondsToSelector:@selector(setSeparatorInset:)]){

[cell setSeparatorInset:UIEdgeInsetsZero];

}

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}#import "MyInfoCenterViewController.h"

#import "ProfileImageCell.h"

#import "BJSettingViewController.h"

#import "UIImageView+EMWebCache.h"

#import "BJMyAttentionViewController.h"

#import "BJBalanceViewController.h"

#import "BJMyEditingInfomationViewController.h"//编辑信息

#import "HYDEvaluationVC.h"//我的评价

#import "BJDriverInfoViewController.h"

#import "MyCustomerOrderMansgeViewController.h" // 订单管理

#import "MyStatementVC.h"//我的报表

#import "BJMyDraftVC.h"//我的草稿

@interface MyInfoCenterViewController ()<UITableViewDataSource, UITableViewDelegate, UIActionSheetDelegate,UIImagePickerControllerDelegate>

@property (nonatomic, strong) UITableView \*myTab;

@property (nonatomic, strong) NSArray \*cellInfoArr;

@property (nonatomic, strong) NSArray \*cellImageNameArr;

@property (nonatomic, strong) NSMutableDictionary \*cacheDict;

@property (nonatomic, strong) NSMutableDictionary \*paramDict;

@property (copy, nonatomic) NSString \*filePath;

@property (nonatomic, strong) UIActionSheet \*myActionSheet;

@property (nonatomic, copy) NSString \*urlImage;//上传图片的URl

@property (nonatomic, copy) NSString \*headUrlImage;//本地头像的URl

@property (nonatomic, strong) NSMutableDictionary \*headImageData;

@property (nonatomic, strong) UILabel \*driverStarImage;

@property (nonatomic, copy) NSString \*myNameStr;

@property (nonatomic, copy) NSString \*myHeaderImgStr;

@property (nonatomic, strong)UIImageView \*headerImage;//头像

@property (nonatomic, strong)UILabel \*driverNameLab;//姓名

@property (nonatomic, assign)CGFloat driverNameLabWidth;

@property (nonatomic, strong)UILabel \*jionInTimeLab;//加入时间

@property (nonatomic, strong)UILabel \*realNameLab;//实名

@property (nonatomic, strong)UILabel \*typeLab;//类别

@property (nonatomic, strong)UILabel \*oneLab;//订单

@property (nonatomic, strong)UILabel \*twoLab;//余额

@property (nonatomic, strong)UILabel \*threeLab; //评价

@property (nonatomic, strong)UILabel \*fourLab;//关注

@property (nonatomic, copy) NSString \*type;

@property (nonatomic, copy) NSString \*Atype;

@property (nonatomic, copy) NSString \*nickName;//用户头像

@property (nonatomic, copy) NSString \*avatarUrl;//用户昵称

@end

@implementation MyInfoCenterViewController

- (instancetype)init

{

self = [super init];

if (self) {

cacheDict = [[NSMutableDictionary alloc] init];

headImageData = [[NSMutableDictionary alloc] init];

}

return self;

}

- (void)viewWillAppear:(BOOL)animated {

[super viewWillAppear:animated];

self.tabBarController.tabBar.hidden = NO;

//昵称头像

\_nickName = [[NSUserDefaults standardUserDefaults] objectForKey:@"name"];

\_avatarUrl = [[NSUserDefaults standardUserDefaults] objectForKey:@"headimg"];

[self getPersonInfo];

}

- (void)viewWillDisappear:(BOOL)animated {

[super viewWillDisappear:animated];

self.tabBarController.tabBar.hidden = YES;

}

- (void)viewDidAppear:(BOOL)animated {

[super viewDidAppear:animated];

self.tabBarController.tabBar.hidden = NO;

}

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.hidesBottomBarWhenPushed = YES;

self.navigationController.navigationBar.barTintColor = kNavigationBarColor;

[self.navigationController.navigationBar setTitleTextAttributes:

@{NSFontAttributeName:[UIFont systemFontOfSize:19],

NSForegroundColorAttributeName:[UIColor whiteColor]}];

self.view.backgroundColor = [UIColor whiteColor];

self.myTab = [[UITableView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight - 64) style:UITableViewStyleGrouped];

self.myTab.scrollEnabled = NO;

self.myTab.delegate = self;

self.myTab.dataSource = self;

[self.view addSubview:\_myTab];

[self setDriverHeader];

}

[super didReceiveMemoryWarning];

}

- (void)setDriverHeader {

UIView \*backView = [[UIView alloc] init];

backView.frame = CGRectMake(0, 0, kDeviceWidth, 135\*UI\_H\_SCALE);

backView.backgroundColor = [UIColor whiteColor];

[self.view addSubview:backView];

UIButton \*headerBtn = [UIButton buttonWithType:UIButtonTypeCustom];

headerBtn.frame = CGRectMake(0, 0, kDeviceWidth, 80\*UI\_H\_SCALE);

[headerBtn addTarget:self action:@selector(pushHeader:) forControlEvents:UIControlEventTouchUpInside];

[backView addSubview:headerBtn];

\_headerImage = [[UIImageView alloc] init];

\_headerImage.frame = CGRectMake(15, 15\*UI\_H\_SCALE, 50\*UI\_H\_SCALE, 50\*UI\_H\_SCALE);

\_headerImage.layer.cornerRadius = 25\*UI\_H\_SCALE;

\_headerImage.image = [UIImage imageNamed:@"0804\_touxiang"];

\_headerImage.layer.masksToBounds = YES;

[headerBtn addSubview:\_headerImage];

\_driverNameLab = [[UILabel alloc] init];

\_driverNameLab.font = [UIFont systemFontOfSize:13];

[headerBtn addSubview:\_driverNameLab];

\_realNameLab = [[UILabel alloc] init];

\_realNameLab.layer.masksToBounds = YES;

\_realNameLab.text = @"实";

\_realNameLab.textAlignment = NSTextAlignmentCenter;

\_realNameLab.textColor = [UIColor whiteColor];

\_realNameLab.font = [UIFont systemFontOfSize:13];

\_realNameLab.layer.cornerRadius = 2;

[headerBtn addSubview:\_realNameLab];

\_typeLab = [[UILabel alloc] init];

\_typeLab.text = @"车";

\_typeLab.textAlignment = NSTextAlignmentCenter;

\_typeLab.textColor = [UIColor whiteColor];

\_typeLab.font = [UIFont systemFontOfSize:13];

\_typeLab.layer.masksToBounds = YES;

\_typeLab.layer.cornerRadius = 2;

\_typeLab.backgroundColor = [UIColor colorWithRed:24 / 255.0 green:205 / 255.0 blue:186 / 255.0 alpha:1];

[headerBtn addSubview:\_typeLab];

//星星

UIView \*startView = [[UIView alloc] init];

startView.frame = CGRectMake(50\*UI\_H\_SCALE+25, 30\*UI\_H\_SCALE, 250\*UI\_H\_SCALE, 20\*UI\_H\_SCALE);

[headerBtn addSubview:startView];

\_driverStarImage = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 250\*UI\_H\_SCALE, 20\*UI\_H\_SCALE)];

\_driverStarImage.font = [UIFont systemFontOfSize:13];

[startView addSubview:\_driverStarImage];

//加入时间

\_jionInTimeLab = [[UILabel alloc] init];

\_jionInTimeLab.frame = CGRectMake(50\*UI\_H\_SCALE+25, 55\*UI\_H\_SCALE, 300\*UI\_H\_SCALE, 20\*UI\_H\_SCALE);

\_jionInTimeLab.font = [UIFont systemFontOfSize:12];

[headerBtn addSubview:\_jionInTimeLab];

UIView \*hengView = [[UIView alloc]initWithFrame:CGRectMake(0, 80\*UI\_H\_SCALE, kDeviceWidth, 0.5)];

hengView.backgroundColor = RGBACOLOR(215, 217, 217, 1);

[headerBtn addSubview:hengView];

NSArray \*titleArr = @[@"订单记录",@"我的余额",@"我的评价",@"我的关注"];

for (NSInteger i=0; i<4; i++) {

UIButton \*btn = [UIButton buttonWithType:UIButtonTypeCustom];

btn.frame = CGRectMake(kDeviceWidth/4\*i, 80\*UI\_H\_SCALE+0.5, kDeviceWidth/4, 55\*UI\_H\_SCALE-1);

[btn addTarget:self action:@selector(headerBtnClick:) forControlEvents:UIControlEventTouchUpInside];

btn.tag = 15000+i;

[backView addSubview:btn];

if (i!=3) {

UIView \*shuView = [[UIView alloc]initWithFrame:CGRectMake(kDeviceWidth/4-0.5, 7.5\*UI\_H\_SCALE, 0.5, 40\*UI\_H\_SCALE)];

shuView.backgroundColor = RGBACOLOR(215, 217, 217, 1);

[btn addSubview:shuView];

}

UILabel \*oneSmallLab = [[UILabel alloc] init];

oneSmallLab.frame = CGRectMake(0, 30\*UI\_H\_SCALE, kDeviceWidth/4, 16\*UI\_H\_SCALE);

oneSmallLab.text = titleArr[i];

oneSmallLab.font = FONT(13);

oneSmallLab.textColor = FontCOLOR;

oneSmallLab.textAlignment = NSTextAlignmentCenter;

[btn addSubview:oneSmallLab];

UILabel \*label = [[UILabel alloc] init];

label.frame = CGRectMake(0, 10\*UI\_H\_SCALE, kDeviceWidth/4, 16\*UI\_H\_SCALE);

label.textColor = FontCOLOR;

label.font = FONT(13);

label.textAlignment = NSTextAlignmentCenter;

[btn addSubview:label];

if (i==0) {

oneLab = label;

}else if (i==1){

twoLab = label;

}else if (i==2){

threeLab = label;

}else if (i==3){

fourLab = label;

}

}

self.myTab.tableHeaderView = backView;

}

- (void)headerBtnClick:(UIButton \*)btn

{

if (btn.tag == 15000) {

[self seeOrder:btn];

}else if (btn.tag == 15001){

[self seeYe:btn];

}else if (btn.tag == 15002){

[self seePJ:btn];

}else if (btn.tag == 15003){

[self seeGZ:btn];

}

}

- (void)pushHeader:(UIButton \*)sender1 {

NSLog(@"头像！！！！！");

BJMyEditingInfomationViewController \*myEditingInfoVC = [[BJMyEditingInfomationViewController alloc] init];

myEditingInfoVC.myOldNameStr = \_myNameStr;

myEditingInfoVC.myHeaderImg = \_myHeaderImgStr;

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

}

- (void)seeOrder:(UIButton \*)sender2 {

NSLog(@"订单！！！！！");

//订单管理

MyCustomerOrderMansgeViewController \*orderManagerVC = [[MyCustomerOrderMansgeViewController alloc] init];

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

}

- (void)seeYe:(UIButton \*)sender3 {

NSLog(@"余额！！！！！");

//我的余额

BJBalanceViewController \*balanceVC =[[BJBalanceViewController alloc] init];

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

}

- (void)seePJ:(UIButton \*)sender4 {

NSLog(@"评价！！！！！");

HYDEvaluationVC \*myEvaluateVC = [[HYDEvaluationVC alloc] init];

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

}

- (void)seeGZ:(UIButton \*)sender5 {

NSLog(@"关注！！！！！");

//我的关注

BJMyAttentionViewController \*myAttentionVC = [[BJMyAttentionViewController alloc] init];

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

}

#pragma mark - UITableViewDataSource

- (NSInteger)numberOfSectionsInTableView:(UITableView \*)tableView

{

return 2;

}

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section

{

if (section == 0) {

return 3;

} else {

return 1;

}

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

ProfileImageCell \*cell = [tableView dequeueReusableCellWithIdentifier:@"ID"];

if (cell == nil) {

cell = [[NSBundle mainBundle] loadNibNamed:@"ProfileImageCell" owner:self options:nil].lastObject;

}

cell.selectionStyle = UITableViewCellSelectionStyleNone;

cell.profileCellLabel.text = \_cellInfoArr[indexPath.section][indexPath.row];

cell.profileCellLabel.textColor = FontCOLOR;

NSString \*imageName = \_cellImageNameArr[indexPath.section][indexPath.row];

cell.profileCellImage.image = [UIImage imageNamed:imageName];

if (indexPath.section == 0 && indexPath.row == 0) {

cell.certificationLabel.text = \_cacheDict[@"sh"];

cell.certificationLabel.font = FONT(14);

cell.certificationLabel.textColor = [UIColor grayColor];

} else {

cell.certificationLabel.hidden = YES;

}

return cell;

}

#pragma mark -UITableViewDelegate

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath

{

return 44;

}

- (CGFloat)tableView:(UITableView \*)tableView heightForFooterInSection:(NSInteger)section

{

if (section == 0) {

return 1;

}else {

return 8;

}

}

#pragma mark - 点击cell的方法

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath {

if (indexPath.section == 0 && indexPath.row == 0) {

//实名认证

NSLog(@"实名认证！！！");

[self pushPersonInfoDetail];

} else if (indexPath.section == 0 && indexPath.row == 1) {

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

MyStatementVC \*statementVC = [[MyStatementVC alloc] init];

statementVC.userName = dict[@"username"];

statementVC.statementType = @"1";

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

} else if (indexPath.section == 0 && indexPath.row == 2) {

NSLog(@"申请发票！！");

BJMyDraftVC \*myDraft = [[BJMyDraftVC alloc] init];

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

} else {

NSLog(@"设置！！！");

BJSettingViewController \*settingVc = [[BJSettingViewController alloc] init];

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

}

}

//获取个人信息的方法

- (void)getPersonInfo

{

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSString \*username = [NSString stringWithFormat:@"%@", dict[@"username"]];

NSDictionary \*pamrm = [NSDictionary dictionaryWithObjectsAndKeys:username,@"username", nil];

[BJHttpTool post:Knewapi3GetUserinfo parameters:pamrm success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

cacheDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"货主端\_cacheDict = %@",\_cacheDict);

if ([[\_cacheDict objectForKey:@"errcode"] intValue] == 0) {

[[NSUserDefaults standardUserDefaults] setObject:\_cacheDict[@"sh"] forKey:@"sh"];

[[NSUserDefaults standardUserDefaults] setObject:\_cacheDict[@"waysl"] forKey:@"waysl"];

[[NSUserDefaults standardUserDefaults]synchronize];

if (![\_nickName isEqualToString:\_cacheDict[@"name"]] || ![\_avatarUrl isEqualToString:\_cacheDict[@"headimg"]]) {

//更新缓存的个人信息

[[NSUserDefaults standardUserDefaults] setObject:\_cacheDict[@"name"] forKey:@"name"];//用户昵称

[[NSUserDefaults standardUserDefaults] setObject:\_cacheDict[@"headimg"] forKey:@"headimg"]; //用户头像

NSString \*userOpenId = dict[@"username"];// 用户环信ID

NSString \*nickName = \_cacheDict[@"name"];// 用户昵称

NSString \*avatarUrl = \_cacheDict[@"headimg"];// 用户头像

[UserCacheManager saveInfo:userOpenId imgUrl:avatarUrl nickName:nickName];

}

type = \_cacheDict[@"type"];

Atype = \_cacheDict[@"Atype"];

NSString \*checkStr = \_cacheDict[@"sh"];

if (![checkStr isEqualToString:@"已审核"]) {

realNameLab.backgroundColor = [UIColor darkGrayColor];

} else {

realNameLab.backgroundColor = [UIColor colorWithRed:250 / 255.0 green:163 / 255.0 blue:26 / 255.0 alpha:1];

}

if ([\_Atype isEqualToString:@"客户"]) {

typeLab.text = @"企";

}

myHeaderImgStr = \_cacheDict[@"headimg"];

NSURL \*url = [NSURL URLWithString:\_cacheDict[@"headimg"]];

if (url == nil) {

headerImage.image = [UIImage imageNamed:@"default\_pic"];

}

else{

[\_headerImage sd\_setImageWithURL:url placeholderImage:[UIImage imageNamed:@"default\_pic"]];

}

if ([\_cacheDict[@"name"] length] == 0) {

myNameStr = @"未认证用户";

} else {

myNameStr = \_cacheDict[@"name"];

}

//姓名的宽度

driverNameLabWidth = [self creatLabelwideWithText:\_myNameStr WithFont:[UIFont systemFontOfSize:14]];

driverNameLab.text = \_myNameStr;

driverNameLab.frame = CGRectMake(80, 10, \_driverNameLabWidth, 20);

realNameLab.frame = CGRectMake(\_driverNameLabWidth + 90, 10, 20, 20);

typeLab.frame = CGRectMake(\_driverNameLabWidth + 113, 10, 20, 20);

NSString \*jionTime = [NSString stringWithFormat:@"%@加入,已发布%@条货源", \_cacheDict[@"ttime"], \_cacheDict[@"fbsl"]];

NSRange rang1 = [jionTime rangeOfString:@"布"];

NSRange range2 = [jionTime rangeOfString:@"条"];

jionInTimeLab.attributedText = [self attributedStringWithStr:jionTime font:12 range:NSMakeRange(rang1.location+1, range2.location-rang1.location-1)];

NSString \*oneStr = [NSString stringWithFormat:@"%@单", \_cacheDict[@"ddsl"]];

oneLab.attributedText = [self attributedStringWithStr:oneStr font:16 range:NSMakeRange(0, oneStr.length-1)];

NSString \*moneyStr = \_cacheDict[@"ye"];

int money = [moneyStr intValue];

NSString \*twoStr = [NSString stringWithFormat:@"%d元", money];

twoLab.attributedText = [self attributedStringWithStr:twoStr font:16 range:NSMakeRange(0, twoStr.length-1)];

NSString \*thresStr = [NSString stringWithFormat:@"%@条", \_cacheDict[@"pjsl"]];

threeLab.attributedText = [self attributedStringWithStr:thresStr font:16 range:NSMakeRange(0, thresStr.length-1)];

NSString \*fourStr =[NSString stringWithFormat:@"%@人", \_cacheDict[@"gzsl"]];

fourLab.attributedText = [self attributedStringWithStr:fourStr font:16 range:NSMakeRange(0, fourStr.length-1)];

NSString \*pf = \_cacheDict[@"pf"];

driverStarImage.text = [NSString stringWithFormat:@"总体评分：%@分",pf];

}

}

[self.myTab reloadData];

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

//[lbxManager showHudViewLabelText:@"获取个人信息失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

//[lbxManager showHUDAction:1 showView:self.view];

}];

}

- (NSAttributedString \*)attributedStringWithStr:(NSString \*)str font:(CGFloat)fontSize range:(NSRange)range

{

NSMutableAttributedString \*richText = [[NSMutableAttributedString alloc] initWithString:str];

[richText addAttribute:NSFontAttributeName value:FONT(fontSize) range:range];//设置字体大小

[richText addAttribute:NSForegroundColorAttributeName value:MineTextColor range:range];//设置字体颜色

return richText;

}

- (void)pushPersonInfoDetail {

BJDriverInfoViewController \*driverInfoVc = [[BJDriverInfoViewController alloc] init];

driverInfoVc.InfoDict = \_cacheDict;

driverInfoVc.type = \_type;

driverInfoVc.Atype = \_Atype;

[driverInfoVc initCheckCarBlock:^{

}];

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

}

// 创建Label

- (UILabel \*)creatLabelWithName:(NSString \*)name andFrame:(CGRect)frame {

UILabel \*label = [[UILabel alloc] init];

label.frame = frame;

label.text = name;

label.textColor = [UIColor blackColor];

label.textAlignment = NSTextAlignmentCenter;

label.font = [UIFont systemFontOfSize:17];

return label;

}

-(CGFloat)creatLabelwideWithText:(NSString \*)textStr WithFont:(UIFont \*)sysFont{

UILabel \*label = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 1000, 30)];

UIFont \*font = sysFont;

[label setFont:font];

[label setNumberOfLines:0];

[label setBackgroundColor:[UIColor clearColor]];

NSString \*text = textStr;

CGSize size ;

NSDictionary \*dic = @{NSFontAttributeName:font};

size = [text boundingRectWithSize:CGSizeMake(99999.0, 30.0f) options:NSStringDrawingUsesLineFragmentOrigin|NSStringDrawingUsesFontLeading attributes:dic context:nil].size;

CGRect rect = label.frame;

rect.size = size;

//NSLog(@"==================================");

[label setFrame:rect];

[label setText:text];

return rect.size.width;

}#import "MyCustomerHomeMessageViewController.h"

#import "MyHomeMessageCell.h"

@interface MyCustomerHomeMessageViewController ()<UITableViewDataSource, UITableViewDelegate>

@property (nonatomic, strong) UITableView \*myCustomerHomeMessageTab;

@property (nonatomic, strong) MyHomeMessageCell \*myCell;

@property (nonatomic, strong) NSArray \*titleArr;

@property (nonatomic, strong) NSArray \*titleImageArr;

@property (nonatomic, copy) NSString \*userType;

@property (nonatomic, copy) NSString \*badgeValueStr;

@end

@implementation MyCustomerHomeMessageViewController

- (void)viewWillAppear:(BOOL)animated {

[super viewWillAppear:animated];

self.tabBarController.tabBar.hidden = NO;

id obj = [[NSUserDefaults standardUserDefaults] objectForKey:@"push"];

NSLog(@"%@",obj);

if ([obj isKindOfClass:[NSDictionary class]]) {

NSLog(@"dsada");

NSDictionary \*dict = (NSDictionary\*)obj;

NSLog(@"推送的字典 = %@", dict);

badgeValueStr = [NSString stringWithFormat:@"%@", dict[@"news\_num"]];

// badgeValueStr = [NSString stringWithFormat:@"%@", dict[@"aps"][@"alert"]];

NSLog(@"推送的内容 = %@", \_badgeValueStr);

//[self showBadgeValur:\_badgeValueStr];

} else {

}

- (void)viewWillDisappear:(BOOL)animated {

[super viewWillDisappear:animated];

self.tabBarController.tabBar.hidden = YES;

}

- (void)viewDidAppear:(BOOL)animated {

[super viewDidAppear:animated];

self.tabBarController.tabBar.hidden = NO;

[\_myCustomerHomeMessageTab reloadData];

}

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.hidesBottomBarWhenPushed = YES;

self.navigationController.navigationBar.barTintColor = kNavigationBarColor;

[self.navigationController.navigationBar setTitleTextAttributes:

@{NSFontAttributeName:[UIFont systemFontOfSize:19],

NSForegroundColorAttributeName:[UIColor whiteColor]}];

\_myCustomerHomeMessageTab = [[UITableView alloc] init];

\_myCustomerHomeMessageTab.frame = CGRectMake(0, 0, kDeviceWidth, kDeviceHeight);

\_myCustomerHomeMessageTab.delegate = self;

\_myCustomerHomeMessageTab.dataSource = self;

[self.view addSubview:\_myCustomerHomeMessageTab];

//\_titleArr = @[@"订单", @"车辆", @"系统"];

//\_titleImageArr = @[@"message\_06", @"message\_13", @"message\_15"];

\_titleArr = @[@"系统"];

\_titleImageArr = @[@"message\_15"];

[\_myCustomerHomeMessageTab setSeparatorInset:UIEdgeInsetsMake(0, 0, 0, 0)];

[self setExtraCellLineHidden:\_myCustomerHomeMessageTab];

\_userType = [[NSUserDefaults standardUserDefaults] objectForKey:@"Atype"];

NSLog(@"\_UserType = %@", \_userType);

}

#pragma mark----- UITableViewDataSource

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath {

return 50;

}

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section {

return 1;

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

static NSString \* identifer = @"reuse1";

\_myCell = [tableView dequeueReusableCellWithIdentifier:identifer];

if (!\_myCell) {

myCell = [[MyHomeMessageCell alloc]initWithStyle:UITableViewCellStyleDefault reuseIdentifier:identifer];

}

\_myCell.myTitleLab.text = \_titleArr[indexPath.row];

\_myCell.myMarkImage.image = [UIImage imageNamed:\_titleImageArr[indexPath.row]];

//if (indexPath.row == 0) {

//

//if ([\_badgeValueStr length] != 0) {

// myCell.myTitleLab.fl\_badgeValue = [NSString stringWithFormat:@"%@", \_badgeValueStr];;

//NSLog(@"赋值去吧");

//}

//

//}

return \_myCell ;

}

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath {

////隐藏小红点

//[self.tabBarController.tabBar hideBadgeOnItemIndex:2];

//[[NSUserDefaults standardUserDefaults] setValue:@"" forKey:@"push"];

//[[NSUserDefaults standardUserDefaults] synchronize];

//

//\_myCell = [tableView cellForRowAtIndexPath:indexPath];

//\_myCell.myTitleLab.fl\_badgeValue = @"";

//\_badgeValueStr = @"";

//

//

//LBXManager \*lbxManager = [LBXManager sharedManager];

//[lbxManager showHUDAction:0 showView:self.view];

//

//

//if (indexPath.row == 0) {

//NSLog(@"系统");

//[lbxManager showHudViewLabelText:@"暂无系统信息" detailsLabelText:nil afterDelay:kAfterDelayTime];

//[lbxManager showHUDAction:1 showView:self.view];

//

//

//}

//

//[lbxManager showHUDAction:1 showView:self.view];

[self pushOrderMessageCenter];

}

- (void)pushOrderMessageCenter {

NSLog(@"货主系统！！");

ALERTSHOWREMIND(@"暂无系统消息");

}

-(void)viewDidLayoutSubviews {

if ([\_myCustomerHomeMessageTab respondsToSelector:@selector(setSeparatorInset:)]) {

[\_myCustomerHomeMessageTab setSeparatorInset:UIEdgeInsetsZero];

}

if ([\_myCustomerHomeMessageTab respondsToSelector:@selector(setLayoutMargins:)]){

[\_myCustomerHomeMessageTab setLayoutMargins:UIEdgeInsetsZero];

}

}

-(void)tableView:(UITableView \*)tableView willDisplayCell:(UITableViewCell \*)cell forRowAtIndexPath:(NSIndexPath \*)indexPat{

if ([cell respondsToSelector:@selector(setLayoutMargins:)]) {

[cell setLayoutMargins:UIEdgeInsetsZero];

}

if ([cell respondsToSelector:@selector(setSeparatorInset:)]){

[cell setSeparatorInset:UIEdgeInsetsZero];

}

}

- (void)setExtraCellLineHidden:(UITableView \*)tableView {

UIView \*view = [UIView new];

view.backgroundColor = [UIColor clearColor];

[tableView setTableFooterView:view];

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}#import "BJHomeViewController.h"

#import "SDCycleScrollView.h"

#import "BJDriverHomePageCell.h"

#import "HomeChildController.h" // 未设置常跑路线

#import "HYDNearCarMapVC.h"//附近业务

#import "PublishProducInfotVC.h"//新的发布货源界面

#import "MyCustomerOrderMansgeViewController.h"//订单管理

#import "MyGPSViewController.h"//GPS 定位

#import "BJChatListViewController.h"

#import "MyAddressBookVC.h"

#import "UIView+BadgeValue.h"

#import "HYDFileDatailVC.h"

#import "BJDriverInfoViewController.h" // 认证

#import "BJMyShareVC.h"

#import "BJMsgCount.h"

#define SCREEN\_WIDTH ([[UIScreen mainScreen] bounds].size.width) / 320

@interface BJHomeViewController ()<UICollectionViewDelegate, UICollectionViewDataSource, UICollectionViewDelegateFlowLayout, SDCycleScrollViewDelegate, UINavigationControllerDelegate, BMKLocationServiceDelegate, BMKGeoCodeSearchDelegate>

@property(strong,nonatomic) UICollectionView \*myCustomerHomeCollectionView;

@property (nonatomic, strong) BJDriverHomePageCell \*cell;

@property(nonatomic, strong) UIImageView \*headerImage;

@property (nonatomic, strong) NSArray \*titleArr;

@property (nonatomic, strong) NSArray \*ImageNameArr;

@property (nonatomic, strong) NSMutableArray \*myArray1;

@property (nonatomic, strong) NSMutableArray \*myArray2;

@property (nonatomic, strong) NSMutableDictionary \*myDict1;

@property (nonatomic, strong) NSMutableArray \*ADImageArr;

@property (nonatomic, strong) NSMutableArray \*listArr;

@property (nonatomic, strong) NSString \*myWayStr;

@property (nonatomic, copy) NSString \*userType;

@property (nonatomic, copy) NSString \*myFilePath;

@property (nonatomic, strong)BMKLocationService \*locService;

@property (nonatomic, strong)BMKGeoCodeSearch \*geocodesearch;

@property (nonatomic, assign)double longitude;

@property (nonatomic, assign)double Latitude;

@property (nonatomic, copy) NSString \*provinceStr;

@property (nonatomic, copy) NSString \*cityStr;

@property (nonatomic, assign) NSInteger unreadCount;

@property (nonatomic, assign) NSInteger ispositioning;//是否定位成功0--表示定位成功1---表示没定位成功

@property (nonatomic , strong) BJMsgCount \*msg;

@end

@implementation BJHomeViewController

- (void)viewWillAppear:(BOOL)animated {

[super viewWillAppear:animated];

self.tabBarController.tabBar.hidden = NO;

[UIApplication sharedApplication].statusBarStyle = UIStatusBarStyleLightContent;

[self getDataWithImage];

\_locService.delegate = self;

\_geocodesearch.delegate = self;

id obj = [[NSUserDefaults standardUserDefaults] objectForKey:@"push"];

NSLog(@"%@",obj);

if ([obj isKindOfClass:[NSDictionary class]]) {

[self getJingGuangPushCount];

} else {

}

NSUserDefaults \*user = [NSUserDefaults standardUserDefaults];

[user removeObjectForKey:@"yueche"];

[user synchronize];

NSString \*number = [[NSUserDefaults standardUserDefaults] objectForKey:@"number"];

NSLog(@"number内容 = %@", number);

NOTIFY\_ADD(setupUnreadMessageCount, kSetupUnreadMessageCount);

[self setupUnreadMessageCount];

}

- (void)viewWillDisappear:(BOOL)animated {

[super viewWillDisappear:animated];

self.tabBarController.tabBar.hidden = YES;

\_locService.delegate = nil;

\_geocodesearch.delegate = nil;

}

- (void)viewDidAppear:(BOOL)animated {

[super viewDidAppear:animated];

self.tabBarController.tabBar.hidden = NO;

[\_myCustomerHomeCollectionView reloadData];

}

- (void)viewDidLoad {

[super viewDidLoad];

\_myArray1 = [@[] mutableCopy];

\_myArray2 = [@[] mutableCopy];

\_listArr = [@[] mutableCopy];

\_ADImageArr = [@[] mutableCopy];

\_myDict1 = [[NSMutableDictionary alloc] init];

\_titleArr = @[@"发布货源", @"搜索车源", @"订单管理", @"对话司机", @"卫星定位", @"邀请朋友"];

\_ImageNameArr = @[@"New", @"附近货源@2x", @"orderMag", @"tv\_user\_1", @"GPS", @"邀请朋友"];

//\_ImageNameArr = @[@"tv\_user\_1", @"tv\_user\_2", @"tv\_user\_3", @"tv\_user\_4", @"tv\_user\_5", @"tv\_user\_6"];

// 设置导航控制器的代理为self

self.navigationController.delegate = self;

self.automaticallyAdjustsScrollViewInsets = NO;

self.hidesBottomBarWhenPushed = YES;

self.navigationController.navigationBar.barTintColor = kNavigationBarColor;

[self.navigationController.navigationBar setTitleTextAttributes:

@{NSFontAttributeName:[UIFont systemFontOfSize:19],

NSForegroundColorAttributeName:[UIColor whiteColor]}];

CGFloat Height = 375\*UI\_H\_SCALE;

UICollectionViewFlowLayout \*myFlowLayout = [[UICollectionViewFlowLayout alloc] init];

[myFlowLayout setScrollDirection:UICollectionViewScrollDirectionVertical];

myFlowLayout.headerReferenceSize = CGSizeMake(kDeviceWidth, Height);

myFlowLayout.minimumLineSpacing = 0.01f;

myFlowLayout.minimumInteritemSpacing = 0.01f;

self.myCustomerHomeCollectionView = [[UICollectionView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight-49) collectionViewLayout:myFlowLayout];

self.myCustomerHomeCollectionView.backgroundColor = [UIColor whiteColor];

[self.myCustomerHomeCollectionView registerClass:[BJDriverHomePageCell class] forCellWithReuseIdentifier:@"BJDriverHomePageCell"];

//注册头视图

[self.myCustomerHomeCollectionView registerClass:[UICollectionReusableView class] forSupplementaryViewOfKind:UICollectionElementKindSectionHeader withReuseIdentifier:@"Header"];

self.myCustomerHomeCollectionView.bounces = NO;

self.myCustomerHomeCollectionView.delegate = self;

self.myCustomerHomeCollectionView.dataSource = self;

if (@available(iOS 11.0, \*)) {

self.myCustomerHomeCollectionView.contentInsetAdjustmentBehavior = UIScrollViewContentInsetAdjustmentNever;

self.myCustomerHomeCollectionView.contentInset = UIEdgeInsetsMake(0, 0, 0, 0);

self.myCustomerHomeCollectionView.scrollIndicatorInsets = self.myCustomerHomeCollectionView.contentInset;

}

[self.view addSubview: self.myCustomerHomeCollectionView];

self.msg = [[BJMsgCount alloc]init];

self.msg.msgCount = @"0";

\_geocodesearch = [[BMKGeoCodeSearch alloc]init];

[self getJingGuangPushCount];

NSNotificationCenter \* center = [NSNotificationCenter defaultCenter];

[center addObserver:self selector:@selector(notice:) name:@"DOCUMENTFILE" object:nil];

[center addObserver:self selector:@selector(pushnotice:) name:@"PUSHCONECT" object:nil];

\_locService = [[BMKLocationService alloc] init];

[self.locService startUserLocationService];

}

-(void)pushnotice:(NSNotification \*)sender{

[self getJingGuangPushCount];

}

-(void)notice:(NSNotification \*)sender{

NSLog(@"%@",sender.userInfo);

HYDFileDatailVC \*datail = [[HYDFileDatailVC alloc]init];

datail.path = sender.userInfo[@"file"];

[self presentViewController:datail animated:YES completion:^{

}];

}

- (void)getJingGuangPushCount

{

NSDictionary \*userDict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSDictionary \*dict = [[NSDictionary alloc]initWithObjectsAndKeys:userDict[@"username"],@"username", nil];

[[BJHttpClient shareInstance]postRequestWithUrl:KGetUreadMessage withParam:dict withView:nil withText:nil sucessBlock:^(id data) {

if ([data[@"errcode"] intValue]==0) {

self.msg.msgCount = data[@"mcount"];

}else{

self.msg.msgCount = @"0";

}

[self.myCustomerHomeCollectionView reloadData];

} faileBlock:^(NSString \*error) {

self.msg.msgCount = @"0";

}];

}

-(void)observeValueForKeyPath:(NSString \*)keyPath ofObject:(id)object change:(NSDictionary<NSString \*,id> \*)change context:(void \*)context

{

[self.myCustomerHomeCollectionView reloadData];

[self.msg removeObserver:self forKeyPath:@"msgCount"];

}

// 统计未读消息数

-(void)setupUnreadMessageCount {

NSArray \*conversations = [[EMClient sharedClient].chatManager getAllConversations];

//NSInteger unreadCount = 0;

\_unreadCount = 0;

for (EMConversation \*conversation in conversations) {

unreadCount += conversation.unreadMessagesCount;

}

NSLog(@"\_unreadCount = %ld", (long)\_unreadCount);

}

#pragma mark - 头视图

//执行的 headerView 代理返回 headerView 的高度

- (UICollectionReusableView \*)collectionView:(UICollectionView \*)collectionView viewForSupplementaryElementOfKind:(NSString \*)kind atIndexPath:(NSIndexPath \*)indexPath

{

//如果是头视图

if ([kind isEqualToString:UICollectionElementKindSectionHeader]) {

UICollectionReusableView \*header=[collectionView dequeueReusableSupplementaryViewOfKind:kind withReuseIdentifier:@"Header" forIndexPath:indexPath];

//头视图添加view

[header addSubview:self.headerImage];

return header;

} else {

return nil;

}

}

-(void)addContent {

CGFloat Height = 375\*UI\_H\_SCALE;

UIImageView \*headerImage1=[[UIImageView alloc]init];

headerImage1.contentMode=UIViewContentModeScaleAspectFill;

headerImage1.frame=CGRectMake(0, 0, self.view.frame.size.width, Height);

headerImage1.backgroundColor = [UIColor whiteColor];

headerImage1.userInteractionEnabled = YES;

SDCycleScrollView \*cycleScrollView = [SDCycleScrollView cycleScrollViewWithFrame:CGRectMake(0, 0, kDeviceWidth, Height) imageURLStringsGroup:\_myArray2]; // 模拟网络延时情景

cycleScrollView.pageControlAliment = SDCycleScrollViewPageContolAlimentRight;

cycleScrollView.delegate = self;

cycleScrollView.backgroundColor = [UIColor clearColor];

//cycleScrollView.titlesGroup = titles;

cycleScrollView.dotColor = kOrangeColor; // 自定义分页控件小圆标颜色

cycleScrollView.placeholderImage = [UIImage imageNamed:@"banner"];

[headerImage1 addSubview:cycleScrollView];

//--- 模拟加载延迟

dispatch\_after(dispatch\_time(DISPATCH\_TIME\_NOW, (int64\_t)(0.5 \* NSEC\_PER\_SEC)), dispatch\_get\_main\_queue(), ^{

cycleScrollView.imageURLStringsGroup = \_myArray2;

});

self.headerImage = headerImage1;

[self.myCustomerHomeCollectionView reloadData];

#pragma mark - UICollectionViewDataSource

- (NSInteger)collectionView:(UICollectionView \*)collectionView numberOfItemsInSection:(NSInteger)section{

return 6;

}

- (UICollectionViewCell \*)collectionView:(UICollectionView \*)collectionView cellForItemAtIndexPath:(NSIndexPath \*)indexPath{

\_cell = [collectionView dequeueReusableCellWithReuseIdentifier:@"BJDriverHomePageCell" forIndexPath:indexPath];

\_cell.imageView.image = [UIImage imageNamed:\_ImageNameArr[indexPath.row]];

\_cell.descLabel.text = \_titleArr[indexPath.row];

NSInteger allCount =\_unreadCount + [self.msg.msgCount integerValue];

if (indexPath.row == 3) {

if (allCount != 0) {

cell.imageView.fl\_badgeValue = [NSString stringWithFormat:@"%ld", (long)allCount];

NSLog(@"赋值去吧");

}

} else {

cell.imageView.fl\_badgeValue = @"0";

}

return \_cell;

}

#pragma mark - UICollectionViewDelegateFlowLayout

- (CGSize)collectionView:(UICollectionView \*)collectionView layout:(UICollectionViewLayout\*)collectionViewLayout sizeForItemAtIndexPath:(NSIndexPath \*)indexPath{

//return CGSizeMake((kDeviceWidth - 80) / 3, (kDeviceWidth - 80) / 3 + 20);

float height = kDeviceHeight;

NSLog(@"%f %f%f",(kDeviceWidth-4)/3,(height-49-370\*UI\_H\_SCALE-3)/2,UI\_H\_SCALE);

return CGSizeMake((kDeviceWidth-4)/3, (height-49-370\*UI\_H\_SCALE-3)/2);

}

- (UIEdgeInsets)collectionView:(UICollectionView \*)collectionView layout:(UICollectionViewLayout\*)collectionViewLayout insetForSectionAtIndex:(NSInteger)section{

return UIEdgeInsetsMake(1, 1, 1, 1);

}

#pragma mark - UICollectionViewDelegate

- (void)collectionView:(UICollectionView \*)collectionView didSelectItemAtIndexPath:(NSIndexPath \*)indexPath {

if (indexPath.row == 0) {

NSString \*examineState = [[NSUserDefaults standardUserDefaults] objectForKey:@"sh"];

if ([examineState isEqualToString:@"未审核"]) {

[self dealUnexamine];

} else if ([examineState isEqualToString:@"审核中"]) {

ALERTSHOWREMIND(@"账户信息在审核中,请稍后");

return;

} else {

NSLog(@"发布货源");

PublishProducInfotVC \*publishVC = [[PublishProducInfotVC alloc] init];

//此处传空。方便草稿重发 和 失效重发的处理!

NSUserDefaults \*user = [NSUserDefaults standardUserDefaults];

NSString \*orderid = [user valueForKey:@"orderid"];

publishVC.orderID = [[orderid componentsSeparatedByString:@"-"] firstObject];

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

}

} else if (indexPath.row == 1) {

NSLog(@"附近车源");

CLAuthorizationStatus status = [CLLocationManager authorizationStatus];

if (kCLAuthorizationStatusDenied == status || kCLAuthorizationStatusRestricted == status) {

NSLog(@"关闭的");

ALERTSHOWREMIND(@"为了更好的体验,请到设置->隐私->定位服务中开启!【好运道APP】定位服务,已便获取附近信息");

return;

} else {

NSLog(@"开启的");

if (\_ispositioning == 0) {

//跳转

HYDNearCarMapVC \*bussinessVC = [[HYDNearCarMapVC alloc] init];

bussinessVC.provice = \_provinceStr;

bussinessVC.city = \_cityStr;

bussinessVC.lat = \_Latitude;

bussinessVC.lng = \_longitude;

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

} else {

//继续获取位置信息

NSLog(@"继续获取位置信息");

HYDNearCarMapVC \*bussinessVC = [[HYDNearCarMapVC alloc] init];

bussinessVC.provice = @"";

bussinessVC.city = @"";

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

}

}

}else if (indexPath.row == 2) {

NSLog(@"订单管理");

MyCustomerOrderMansgeViewController \*myCustomerOrderManageVC = [[MyCustomerOrderMansgeViewController alloc] init];

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

}else if (indexPath.row == 3) {

NSLog(@"对话司机");

[[NSUserDefaults standardUserDefaults] setValue:@"" forKey:@"push"];

[[NSUserDefaults standardUserDefaults] synchronize];

cell = (BJDriverHomePageCell \*)[collectionView cellForItemAtIndexPath:indexPath];

BJChatListViewController \*ChatListVC = [[BJChatListViewController alloc] init];

ChatListVC.badgeValue = self.msg;

ChatListVC.userType = @"我是货主";

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

}else if (indexPath.row == 4) {

NSLog(@"CPS定位");

MyGPSViewController \*myLocationVC = [[MyGPSViewController alloc] init];

myLocationVC.isCustomer = YES;

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

}else if (indexPath.row == 5) {

NSLog(@"邀请朋友");

BJMyShareVC \*vc = [[BJMyShareVC alloc] init];

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

}

}

//处理未审核时点击 发布货源

- (void) dealUnexamine {

UIAlertView \*myShowView = [[UIAlertView alloc] initWithTitle:@"" message:@"请进行实名认证" delegate:self cancelButtonTitle:@"立即设置" otherButtonTitles:@"下次设置", nil];

[myShowView show];

}

//Atype-----公司，个人

//type----货主，车主

- (void)alertView:(UIAlertView \*)alertView clickedButtonAtIndex:(NSInteger)buttonIndex {

switch (buttonIndex) {

case 0: {

NSLog(@"立即设置");

NSString \*AtypeStr = [[NSUserDefaults standardUserDefaults] objectForKey:@"Atype"];

NSString \*type = [[NSUserDefaults standardUserDefaults] objectForKey:@"type"];

BJDriverInfoViewController \*myInfoVC = [[BJDriverInfoViewController alloc] init];

myInfoVC.Atype = AtypeStr;

myInfoVC.type = type;

[myInfoVC initCheckCarBlock:^{

}];

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

}

break;

case 1: {

NSLog(@"下次设置");

// 不做处理

}

break;

default:

break;

}

}

#pragma mark - SDCycleScrollViewDelegate

- (void)cycleScrollView:(SDCycleScrollView \*)cycleScrollView didSelectItemAtIndex:(NSInteger)index {

NSLog(@"点击了 = %ld", (long)index);

NSDictionary \*adDic = \_myArray1[index];

if (adDic==nil) {

return;

}

NSString \*linkUrl = adDic[@"tourl"];

if (linkUrl && ![linkUrl isEqualToString:@""]) {

NSLog(@"\*\*\*\*linkUrl = %@",linkUrl);

#pragma mark-----暂时不让跳转

}

}

#pragma mark----数据

- (void)getDataWithImage {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

NSMutableDictionary \*parmDict = [[NSMutableDictionary alloc] init];

[parmDict setValue:@"1" forKey:@"type"];

[BJHttpTool Post:KGetIndexImg parameters:parmDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*rootDict = [[NSMutableDictionary alloc] init];

rootDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"rootDict = %@",rootDict);

if ([[rootDict objectForKey:@"errcode"] intValue] == 0) {

myArray1 = [NSMutableArray arrayWithArray:rootDict[@"list"]];

NSLog(@"\_myArray1 = %@", \_myArray1);

[\_myArray2 removeAllObjects];

for (int i = 0; i < \_myArray1.count; i++) {

myDict1 = \_myArray1[i];

[\_myArray2 addObject:\_myDict1[@"imgurl"]];

}

//添加头视图的内容

[self addContent];

}

}

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

}];

}

//处理位置坐标更新

- (void)didUpdateBMKUserLocation:(BMKUserLocation \*)userLocation {

// [self.locService stopUserLocationService];停止定位

NSLog(@"当前的坐标是: 经度%f,纬度%f",userLocation.location.coordinate.latitude,userLocation.location.coordinate.longitude);

\_longitude = userLocation.location.coordinate.longitude;

\_Latitude = userLocation.location.coordinate.latitude;

[self reverseGeocode];

}

- (void)reverseGeocode {

CLLocationCoordinate2D pt = (CLLocationCoordinate2D){0, 0};

pt = (CLLocationCoordinate2D){\_Latitude, \_longitude};

NSLog(@"%f, %f", pt.latitude, pt.longitude);

BMKReverseGeoCodeOption \*reverseGeocodeSearchOption = [[BMKReverseGeoCodeOption alloc]init];

reverseGeocodeSearchOption.reverseGeoPoint = pt;

BOOL flag = [\_geocodesearch reverseGeoCode:reverseGeocodeSearchOption];

if(flag)

{

NSLog(@"反geo检索发送成功");

}

else

{

NSLog(@"反geo检索发送失败");

}

}

#pragma mark------获取定位信息

-(void) onGetReverseGeoCodeResult:(BMKGeoCodeSearch \*)searcher result:(BMKReverseGeoCodeResult \*)result errorCode:(BMKSearchErrorCode)error

{

if (error == 0) {

//获取定位信息成功

ispositioning = 0;

BMKPointAnnotation\* item = [[BMKPointAnnotation alloc]init];

item.coordinate = result.location;

item.title = result.address;

NSLog(@"我要的地址=%@", result.address);

NSString \*pstr = result.addressDetail.province;

NSString \*pstr1 = result.addressDetail.city;

NSString \*pstr2 = result.addressDetail.district;

provinceStr = result.addressDetail.province;

cityStr = result.addressDetail.city;

NSLog(@"组合地址 = %@- %@ - %@", pstr, pstr1, pstr2);

} else {

NSLog(@"设么鬼 = %u", error);

//获取定位信息失败

ispositioning = 1;

return;

}

}

#pragma mark - UINavigationControllerDelegate

// 将要显示控制器

- (void)navigationController:(UINavigationController \*)navigationController willShowViewController:(UIViewController \*)viewController animated:(BOOL)animated {

// 判断要显示的控制器是否是自己

BOOL isShowHomePage = [viewController isKindOfClass:[self class]];

[self.navigationController setNavigationBarHidden:isShowHomePage animated:YES];

}- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString \*)reuseIdentifier {

self = [super initWithStyle:style reuseIdentifier:reuseIdentifier];

if (self) {

[self.contentView addSubview:self.bigView];

[self.bigView addSubview:self.timeLabel];

[self.bigView addSubview:self.priceLab];

[self.bigView addSubview:self.underline];

[self.bigView addSubview:self.startLab];

[self.bigView addSubview:self.smallImage];

[self.bigView addSubview:self.arriveLab];

[self.bigView addSubview:self.goodsDetailLab];

[self.bigView addSubview:self.personImage];

[self.bigView addSubview:self.nameLab];

[self.bigView addSubview:self.telBtn];

[self.bigView addSubview:self.companyLab];

[self.bigView addSubview:self.distanceLab];

}

return self;

}

- (UIView \*)bigView {

if (!\_bigView) {

bigView = [[UIView alloc] init];

bigView.frame = CGRectMake(0, 0, kDeviceWidth, 110);

bigView.backgroundColor = [UIColor whiteColor];

}

return \_bigView;

}

- (UILabel \*)timeLabel {

if (!\_timeLabel) {

timeLabel = [[UILabel alloc] init];

timeLabel.frame = CGRectMake(10, 0, 200, 20);

timeLabel.text = @"装货时间:2012-7-22";

timeLabel.font = [UIFont systemFontOfSize:13];

}

return \_timeLabel;

}

- (UILabel \*)priceLab {

if (!\_priceLab) {

priceLab = [[UILabel alloc] init];

priceLab.frame = CGRectMake(kDeviceWidth - 120, 0, 110, 20);

priceLab.text = @"最低价:555455";

priceLab.textAlignment = NSTextAlignmentRight;

priceLab.font = [UIFont systemFontOfSize:13];

}

return \_priceLab;

}

- (UIView \*)underline {

if (!\_underline) {

underline = [[UIView alloc] init];

underline.frame = CGRectMake(0, 21, kDeviceWidth, 1);

underline.backgroundColor = MyCellColor;

}

return \_underline;

}

- (UILabel \*)startLab {

if (!\_startLab) {

startLab = [[UILabel alloc] init];

startLab.frame = CGRectMake(10, 21, 110, 30);

startLab.text = @"上海市";

startLab.font = [UIFont boldSystemFontOfSize:15];

// startLab.font = [UIFont systemFontOfSize:15];

}

return \_startLab;

}

- (UIImageView \*)smallImage {

if (!\_smallImage) {

smallImage = [[UIImageView alloc] init];

smallImage.frame = CGRectMake(120, 28, 20, 15);

smallImage.image = [UIImage imageNamed:@"to"];

smallImage.contentMode = UIViewContentModeScaleAspectFit;

}

return \_smallImage;

}

- (UILabel \*)arriveLab {

if (!\_arriveLab) {

arriveLab = [[UILabel alloc] init];

arriveLab.frame = CGRectMake(141, 21, 120, 30);

arriveLab.text = @"重庆市万盛区";

arriveLab.font = [UIFont boldSystemFontOfSize:15];

// arriveLab.font = [UIFont systemFontOfSize:15];

}

return \_arriveLab;

}

- (UILabel \*)goodsDetailLab {

if (!\_goodsDetailLab) {

goodsDetailLab = [[UILabel alloc] init];

goodsDetailLab.frame = CGRectMake(10, 51, 210, 20);

goodsDetailLab.text = @"铺货/12.5米/40吨";

goodsDetailLab.font = [UIFont systemFontOfSize:13];

}

return \_goodsDetailLab;

}

- (UIImageView \*)personImage {

if (!\_personImage) {

personImage = [[UIImageView alloc] init];

personImage.frame = CGRectMake(10, 73, 35, 35);

// personImage.backgroundColor = [UIColor purpleColor];

personImage.layer.masksToBounds = YES;

personImage.layer.cornerRadius = 17.5;

}

return \_personImage;

}

- (UILabel \*)nameLab {

if (!\_nameLab) {

nameLab = [[UILabel alloc] init];

nameLab.frame = CGRectMake(50, 80, 45, 20);

nameLab.text = @"干贝司";

nameLab.textAlignment = NSTextAlignmentCenter;

nameLab.font = [UIFont systemFontOfSize:13];

// nameLab.backgroundColor = [UIColor greenColor];

}

return \_nameLab;

}

- (UILabel \*)distanceLab {

if (!\_distanceLab) {

distanceLab = [[UILabel alloc] init];

distanceLab.frame = CGRectMake(102, 80, 20, 20);

distanceLab.text = @"实";

distanceLab.textColor = [UIColor whiteColor];

distanceLab.layer.masksToBounds = YES;

distanceLab.layer.cornerRadius = 5;

distanceLab.textAlignment = NSTextAlignmentCenter;

distanceLab.backgroundColor = [UIColor colorWithRed:251/255.0 green:163/255.0 blue:9/255.0 alpha:1];

distanceLab.font = [UIFont systemFontOfSize:12];

}

return \_distanceLab;

}

- (UILabel \*)companyLab {

if (!\_companyLab) {

companyLab = [[UILabel alloc] init];

companyLab.frame = CGRectMake(125, 80, 20, 20);

companyLab.text = @"企";

companyLab.textColor = [UIColor whiteColor];

companyLab.textAlignment = NSTextAlignmentCenter;

companyLab.layer.masksToBounds = YES;

companyLab.layer.cornerRadius = 5;

companyLab.backgroundColor = [UIColor colorWithRed:25/255.0 green:202/255.0 blue:185/255.0 alpha:1];

companyLab.font = [UIFont systemFontOfSize:12];

}

return \_companyLab;

}

- (UIButton \*)telBtn {

if (!\_telBtn) {

telBtn = [UIButton buttonWithType:UIButtonTypeCustom];

telBtn.frame = CGRectMake(kDeviceWidth - 65, 45, 40, 40);

[\_telBtn setImage:[UIImage imageNamed:@"openTel.png"] forState:UIControlStateNormal];

// telBtn.backgroundColor = kOrangeColor;

telBtn.imageView.contentMode = UIViewContentModeScaleAspectFit;

telBtn.layer.masksToBounds = YES;

telBtn.layer.cornerRadius = 20;

telBtn.titleLabel.font = [UIFont systemFontOfSize:14];

//[\_telBtn setTitle:@"竞价" forState:UIControlStateNormal];

[\_telBtn setTintColor:[UIColor whiteColor]];

}

return \_telBtn;

}

#import "HYDNearCarDetailView.h"

#import "UIImageView+AFNetworking.h"

@implementation HYDNearCarDetailView

- (instancetype)initWithFrame:(CGRect)frame

{

self = [super initWithFrame:frame];

if (self) {

self.backgroundColor = [UIColor whiteColor];

[self addSubview:self.cphLabel];

[self addSubview:self.datailLabel];

[self addSubview:self.yuYueBtn];

[self addSubview:self.iconView];

[self addSubview:self.nameLab];

}

return self;

}

- (void)showWithModel:(HYDNearCarModel \*)model

{

self.cphLabel.text = model.cph;

self.datailLabel.text = [NSString stringWithFormat:@"%@:%@%@:%@",model.driver,model.driverpf.length==0?@"暂无评分":model.driverpf,model.username,model.usernamepf.length==0?@"暂无评分":model.usernamepf];

[self.iconView setImageWithURL:[NSURL URLWithString:model.userheadimg] placeholderImage:[UIImage imageNamed:@"default\_pic"]];

self.nameLab.text = model.username;

[UIView animateWithDuration:0.5 animations:^{

self.frame = CGRectMake(0, IS\_IPHONEx? kDeviceHeight-self.frame.size.height-88: kDeviceHeight-self.frame.size.height-64, kDeviceWidth, self.frame.size.height);

}];

}

- (void)close

{

[UIView animateWithDuration:0.5 animations:^{

self.frame = CGRectMake(0, kDeviceHeight, kDeviceWidth, self.frame.size.height);

}];

}

- (void)yuyueBtnClick

{

}

- (UILabel \*)cphLabel

{

if (!\_cphLabel) {

cphLabel = [[UILabel alloc]initWithFrame:CGRectMake(15, 5, 200, 30)];

cphLabel.font = FONT(16);

}

return \_cphLabel;

}

- (UILabel \*)datailLabel

{

if (!\_datailLabel) {

datailLabel = [[UILabel alloc]initWithFrame:CGRectMake(15, 35, kDeviceWidth-30-65, 30)];

datailLabel.font = FONT(16);

}

return \_datailLabel;

}

- (UIButton \*)yuYueBtn

{

if (!\_yuYueBtn) {

yuYueBtn = [UIButton buttonWithType:UIButtonTypeCustom];

yuYueBtn.frame = CGRectMake(kDeviceWidth-95, 35, 60, 30);

yuYueBtn.backgroundColor = kNavigationBarColor;

yuYueBtn.layer.masksToBounds = YES;

yuYueBtn.layer.cornerRadius = 5;

[\_yuYueBtn setTitle:@"约车" forState:UIControlStateNormal];

[\_yuYueBtn addTarget:self action:@selector(yuyueBtnClick) forControlEvents:UIControlEventTouchUpInside];

}

return \_yuYueBtn;

}

- (UIImageView \*)iconView

{

if (!\_iconView) {

iconView = [[UIImageView alloc]initWithFrame:CGRectMake(15, 65, 40, 40)];

iconView.layer.masksToBounds = YES;

iconView.layer.cornerRadius = 20;

}

return \_iconView;

}

- (UILabel \*)nameLab

{

if (!\_nameLab) {

nameLab = [[UILabel alloc]initWithFrame:CGRectMake(70, 65, 100, 40)];

nameLab.font = FONT(16);

}

return \_nameLab;

}

#import "MyCutomerCarsInfoViewController.h"

#import "MyCarsInfoCell.h"

#import "BJAddressTableViewCell.h"

#import "BJRemarkViewCell.h"

#import "ChatViewController.h" //聊天界面

#define leftDistance 24// 左边距

#define labelTopDistance6 // 上边距

#define lineLeftDistance 8

@interface MyCutomerCarsInfoViewController ()<UITableViewDelegate, UITableViewDataSource, UIAlertViewDelegate>

@property (nonatomic,strong) UITableView \*myCustomerCarInfoTab;

@property (nonatomic,retain)NSDictionary \* dataDic;

@property (nonatomic,retain)NSArray \* orderedKeys;

@property (nonatomic, strong) UIView \*bg;

@property (nonatomic, strong)UILabel \*driverNameLab;//姓名

@property (nonatomic, copy) NSString \*myNameStr;

@property (nonatomic, assign)CGFloat driverNameLabWidth;

@property (nonatomic, strong) UIImageView \*myHeaderView;

@property (nonatomic, strong) UILabel \*realNameLab;

@property (nonatomic, strong)UILabel \*typeLab;//类别

@property (nonatomic, strong) UILabel \*jionTimeLab;

@property (nonatomic, strong) UILabel \*priceLab;

@property (nonatomic, strong) UILabel \*labelTitle;

@property (nonatomic, strong) UILabel \*StarImage;

@property (nonatomic, copy) NSString \*telStr;

@property (nonatomic, copy) NSString \*attentStr;

@property (nonatomic, copy) NSString \*attentionStr;

@property (nonatomic, strong) NSMutableDictionary \*myCarInfoDict;

@property (nonatomic, strong) UIButton \*mapBtn;

@property (nonatomic, copy) NSString \*startStr;

@property (nonatomic, copy) NSString \*arriveStr;

@property (nonatomic, strong) UIButton \*attentionBtn;

@property (nonatomic, strong) UIButton \*IMbtn;

@property (nonatomic, strong) UIButton \*phoneBtn;

end

@implementation MyCutomerCarsInfoViewController

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.title = @"车源详情";

\_dataDic = [[NSDictionary alloc] init];

self.myCarInfoDict = [[NSMutableDictionary alloc] init];

\_myCustomerCarInfoTab = [[UITableView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight - 64) style:UITableViewStylePlain];

\_myCustomerCarInfoTab.delegate = self;

\_myCustomerCarInfoTab.dataSource = self;

\_myCustomerCarInfoTab.scrollEnabled = NO;

[self.view addSubview:\_myCustomerCarInfoTab];

[self setExtraCellLineHidden:\_myCustomerCarInfoTab];

NSArray \* aArr = @[@"装车时间:"];

NSArray \* bArr = @[@"出发地点:"];

NSArray \* cArr = @[@"车辆信息:"];

NSArray \* eArr = @[@"备注:"];

self.dataDic = @{@"装车时间":aArr, @"路线":bArr, @"车辆信息":cArr, @"备注:":eArr};

self.orderedKeys = @[@" 时间", @" 线路", @" 车源信息", @" 备注"];

[self initHeaderView];

//请求数据

[self getCarsSourceInfoData];

}

#pragma mark------ UIView

- (void)initHeaderView {

UIView \*bigView = [[UIView alloc] init];

bigView.frame = CGRectMake(0, 0, kDeviceWidth, 70);

//bigView.backgroundColor = [UIColor redColor];

[self.view addSubview:bigView];

self.myHeaderView = [[UIImageView alloc] init];

self.myHeaderView.frame = CGRectMake(10, 5, 60, 60);

self.myHeaderView.layer.masksToBounds = YES;

self.myHeaderView.layer.cornerRadius = 30;

self.myHeaderView.image = [UIImage imageNamed:@"default\_pic"];

[bigView addSubview:self.myHeaderView];

\_driverNameLab = [[UILabel alloc] init];

//\_driverNameLab.frame = CGRectMake(80, 10, 60, 20);

\_driverNameLab.font = [UIFont systemFontOfSize:13];

[bigView addSubview:\_driverNameLab];

\_realNameLab = [[UILabel alloc] init];

//\_realNameLab.frame = CGRectMake(150, 10, 20, 20);

\_realNameLab.layer.masksToBounds = YES;

\_realNameLab.text = @"实";

\_realNameLab.textAlignment = NSTextAlignmentCenter;

\_realNameLab.textColor = [UIColor whiteColor];

\_realNameLab.font = [UIFont systemFontOfSize:13];

\_realNameLab.layer.cornerRadius = 2;

\_realNameLab.backgroundColor = [UIColor colorWithRed:250 / 255.0 green:163 / 255.0 blue:26 / 255.0 alpha:1];

[bigView addSubview:\_realNameLab];

\_typeLab = [[UILabel alloc] init];

//\_typeLab.frame = CGRectMake(173, 10, 20, 20);

\_typeLab.text = @"车";

\_typeLab.textAlignment = NSTextAlignmentCenter;

\_typeLab.textColor = [UIColor whiteColor];

\_typeLab.font = [UIFont systemFontOfSize:13];

\_typeLab.layer.masksToBounds = YES;

\_typeLab.layer.cornerRadius = 2;

\_typeLab.backgroundColor = [UIColor colorWithRed:24 / 255.0 green:205 / 255.0 blue:186 / 255.0 alpha:1];

[bigView addSubview:\_typeLab];

//星星

UIView \*startView = [[UIView alloc] init];

startView.frame = CGRectMake(80, 30, 250, 20);

[bigView addSubview:startView];

\_StarImage = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 250, 20)];

\_StarImage.font = [UIFont systemFontOfSize:13];

[startView addSubview:\_StarImage];

self.jionTimeLab = [[UILabel alloc] init];

self.jionTimeLab.frame = CGRectMake(80, 50, kDeviceWidth - 100, 20);

self.jionTimeLab.font = [UIFont systemFontOfSize:13];

[bigView addSubview:self.jionTimeLab];

\_attentionBtn = [UIButton buttonWithType:UIButtonTypeCustom];

\_attentionBtn.frame = CGRectMake(kDeviceWidth - 120, 22.5, 25, 25);

\_attentionBtn.layer.masksToBounds = YES;

\_attentionBtn.imageView.contentMode = UIViewContentModeScaleAspectFit;

[\_attentionBtn addTarget:self action:@selector(attenBtnAction:) forControlEvents:UIControlEventTouchUpInside];

[bigView addSubview:\_attentionBtn];

\_IMbtn = [UIButton buttonWithType:UIButtonTypeCustom];

\_IMbtn.frame = CGRectMake(kDeviceWidth - 80, 22.5, 25, 25);

\_IMbtn.imageView.contentMode = UIViewContentModeScaleAspectFit;

[\_IMbtn setImage:[UIImage imageNamed:@"IMBtn"] forState:UIControlStateNormal];

[\_IMbtn addTarget:self action:@selector(IMAction:) forControlEvents:UIControlEventTouchUpInside];

[bigView addSubview:\_IMbtn];

NSString \*shType = [[NSUserDefaults standardUserDefaults]stringForKey:@"sh"];

if ([shType isEqualToString:@"已审核"]) {

phoneBtn = [UIButton buttonWithType:UIButtonTypeCustom];

phoneBtn.frame = CGRectMake(kDeviceWidth - 40, 22.5, 25, 25);

phoneBtn.layer.masksToBounds = YES;

phoneBtn.layer.cornerRadius = 12.5;

[\_phoneBtn setImage:[UIImage imageNamed:@"cellBtn11"] forState:UIControlStateNormal];

[\_phoneBtn addTarget:self action:@selector(phoneBtnAction:) forControlEvents:UIControlEventTouchUpInside];

[bigView addSubview:\_phoneBtn];

}

\_myCustomerCarInfoTab.tableHeaderView = bigView;

}

#pragma mark - UITableViewDataSource

- (NSString \*)tableView:(UITableView \*)tableView titleForHeaderInSection:(NSInteger)section {

return self.orderedKeys[section];

}

- (NSInteger)numberOfSectionsInTableView:(UITableView \*)tableView{

return self.dataDic.count;

}

(NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section {

return 1;

}

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath {

if (indexPath.section == 1) {

return 70;

} else if (indexPath.section == 2) {

return 50;

} else if (indexPath.section == 3) {

return 90;

} else {

return 44;

}

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

if (indexPath.section == 0) {

// 时间-货物类型续需求

static NSString \* identifer = @"reuse1";

UITableViewCell \* cell1 = [tableView cellForRowAtIndexPath:indexPath];

if (!cell1) {

cell1 = [[UITableViewCell alloc]initWithStyle:UITableViewCellStyleValue1 reuseIdentifier:identifer];

}

cell1.selectionStyle = UITableViewCellSelectionStyleNone;

cell1.textLabel.textColor = [UIColor blackColor];

cell1.textLabel.font = [UIFont systemFontOfSize:15];

if (indexPath.section == 0) {

cell1.textLabel.text = [NSString stringWithFormat:@"装车时间: %@ %@", \_myCarInfoDict[@"starttime"], \_myCarInfoDict[@"strtime"]];

}

return cell1;

} else if (indexPath.section == 1) {

//路线信息

static NSString \*cellIdentifer = @"addCell";

BJAddressTableViewCell \*addressCell = [tableView dequeueReusableCellWithIdentifier:cellIdentifer];

if (!addressCell) {

addressCell = [[BJAddressTableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:cellIdentifer];

}

addressCell.selectionStyle = UITableViewCellSelectionStyleNone;

addressCell.startImage.image = [UIImage imageNamed:@"start1"];

addressCell.arriveImage.image = [UIImage imageNamed:@"end1"];

startStr = [NSString stringWithFormat:@"%@%@%@%@", \_myCarInfoDict[@"cprovince"], \_myCarInfoDict[@"ccity"], \_myCarInfoDict[@"carea"],\_myCarInfoDict[@"caddress"]];

arriveStr = [NSString stringWithFormat:@"%@%@%@%@", \_myCarInfoDict[@"dprovince"], \_myCarInfoDict[@"dcity"], \_myCarInfoDict[@"darea"],\_myCarInfoDict[@"daddress"]];

addressCell.startLab.text = [NSString stringWithFormat:@"出发地点:%@", \_startStr];

addressCell.arriveLab.text = [NSString stringWithFormat:@"到达地点:%@", \_arriveStr];

return addressCell;

} else if (indexPath.section == 2) {

//车源信息

static NSString \*cellIdentifer1 = @"myCarsInfo";

MyCarsInfoCell \*myCarsInfo = [tableView dequeueReusableCellWithIdentifier:cellIdentifer1];

if (!myCarsInfo) {

myCarsInfo = [[MyCarsInfoCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:cellIdentifer1];

}

myCarsInfo.selectionStyle = UITableViewCellSelectionStyleNone;

myCarsInfo.carInfoLab.font = [UIFont systemFontOfSize:15];

myCarsInfo.carNumberLab.font = [UIFont systemFontOfSize:15];

myCarsInfo.carNumberLab.text = [NSString stringWithFormat:@"%@" , \_myCarInfoDict[@"cph"]];

myCarsInfo.carInfoLab.text = [NSString stringWithFormat:@"%@ %@" , \_myCarInfoDict[@"cartype"], \_myCarInfoDict[@"csize"]];

return myCarsInfo;

} else {

//备注

static NSString \*identifer1 = @"remmarkCell";

BJRemarkViewCell \*remmarkCell = [tableView dequeueReusableCellWithIdentifier:identifer1];

//3.判断是否能够成功取到可重用的cell

if (!remmarkCell) {

//4.如果没有取到,创建一个cell

remmarkCell = [[BJRemarkViewCell alloc]initWithStyle:UITableViewCellStyleDefault reuseIdentifier:identifer1];

}

remmarkCell.selectionStyle = UITableViewCellSelectionStyleNone;

remmarkCell.remarkLab.text = \_myCarInfoDict[@"bz"];

return remmarkCell;

}

}

#pragma mark------- Action

//关注

- (void)attenBtnAction:(UIButton \*)attenSender {

if ([\_attentStr isEqualToString:@"取消关注成功"] ||![\_attentionStr isEqualToString:@"True"]) {

[self setAttentionWithData];

} else {

[self cancleAttentionWithData];

}

}

//IM

- (void)IMAction:(UIButton \*)sender {

NSLog(@"IM ");

ChatViewController \*vc = [[ChatViewController alloc]initWithConversationChatter:self.phoneStr conversationType:EMConversationTypeChat];

vc.title = @"对话信息";

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

}

//电话

- (void)phoneBtnAction:(UIButton \*)sender1 {

NSLog(@"电话!!!! ");

[self openURLToCall];

}

#pragma mark------拨打电话

- (void)openURLToCall {

NSLog(@"货主电话");

NSString \*phoneStr = self.phoneStr;

//防止电话号码 为空时程序崩溃

if ([self.phoneStr isEqual: @""] || self.phoneStr == nil || self.phoneStr.length == 0) {

return ;

} else {

UIAlertView \*alertView = [[UIAlertView alloc] initWithTitle:@"联系车主" message:phoneStr delegate:self cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

[alertView show];

}

}

#pragma mark - UIAlertView的代理

- (void)alertView:(UIAlertView \*)alertView clickedButtonAtIndex:(NSInteger)buttonIndex

{

if (buttonIndex == alertView.cancelButtonIndex) {

NSLog(@"取消");

} else if (buttonIndex == 1){

[self openPhone];

}

}

//打客服电话

- (void)openPhone {

NSMutableString \*str = [[NSMutableString alloc] initWithFormat:@"tel:%@",self.phoneStr];

UIWebView \* callWebview = [[UIWebView alloc] init];

[callWebview loadRequest:[NSURLRequest requestWithURL:[NSURL URLWithString:str]]];

[self.view addSubview:callWebview];

}

#pragma amrk------数据

//关注

- (void)setAttentionWithData {

NSLog(@"已经关注");

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSMutableDictionary \*attentionDict = [[NSMutableDictionary alloc] init];

[attentionDict setValue:\_phoneStr forKey:@"Duser"];

[attentionDict setValue:dict[@"username"] forKey:@"username"];

NSLog(@"已经关注attentionDict:%@", attentionDict);

[BJHttpTool Post:KAttDriver parameters:attentionDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*attentionData = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"attentionData = %@",attentionData);

if ([attentionData[@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"已成功关注" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_follow11"] forState:UIControlStateNormal];

}

if ([[attentionData objectForKey:@"errcode"] intValue] == 1) {

[self cancleAttentionWithData];

NSLog(@"跑到了这里！！！！");

}

}

} failure:^(NSError \*error) {

NSLog(@"error = %@", error);

[lbxManager showHudViewLabelText:@"关注失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

}

//取消关注

- (void)cancleAttentionWithData {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSMutableDictionary \*cancleAttentionDict = [[NSMutableDictionary alloc] init];

[cancleAttentionDict setValue:dict[@"username"] forKey:@"username"];

[cancleAttentionDict setValue:\_phoneStr forKey:@"Duser"];

NSLog(@"取消关注cancleAttentionDict:%@", cancleAttentionDict);

[BJHttpTool Post:KCancelAttDriver parameters:cancleAttentionDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*cancleAttentionData = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"cancleAttentionData = %@",cancleAttentionData);

if ([[cancleAttentionData objectForKey:@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"取消关注" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_unfollow11"] forState:UIControlStateNormal];

attentStr = @"取消关注成功";

}

}

} failure:^(NSError \*error) {

NSLog(@"error = %@", error);

[lbxManager showHudViewLabelText:@"取消关注失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

}

/车源详情

- (void)getCarsSourceInfoData {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSMutableDictionary \*paramDict = [[NSMutableDictionary alloc] init];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

[paramDict setValue:dict[@"username"] forKey:@"username"];

[paramDict setValue:\_carID forKey:@"tid"];

[paramDict setValue:\_phoneStr forKey:@"czuser"];

NSLog(@"所有的参数 = %@", paramDict);

[BJHttpTool Post:KGetcarfbInfoByUser parameters:paramDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*rootDict = [[NSMutableDictionary alloc] init];

rootDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"车源详情rootDict = %@",rootDict);

if ([[rootDict objectForKey:@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"获取车源信息成功" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

self.myCarInfoDict = [NSMutableDictionary dictionaryWithDictionary:rootDict];

NSURL \*url = [NSURL URLWithString:rootDict[@"headimg"]];

if (url == nil) {

myHeaderView.image = [UIImage imageNamed:@"default\_pic"];

}

else{

[\_myHeaderView sd\_setImageWithURL:url placeholderImage:[UIImage imageNamed:@"default\_pic"]];

}

myNameStr = rootDict[@"name"];

//姓名的宽度

driverNameLabWidth = [self creatLabelwideWithText:\_myNameStr WithFont:[UIFont systemFontOfSize:14]];

driverNameLab.text = \_myNameStr;

driverNameLab.frame = CGRectMake(80, 10, \_driverNameLabWidth, 20);

realNameLab.frame = CGRectMake(\_driverNameLabWidth + 90, 10, 20, 20);

typeLab.frame = CGRectMake(\_driverNameLabWidth + 113, 10, 20, 20);

//NSString \*str1 = rootDict[@"fb\_type"];

//if ([str1 isEqualToString:@"公司"]) {

// usersTypeLab.text = @"企";

//} else {

// usersTypeLab.text = @"个";

//}

NSString \*pf = rootDict[@"pf"];

StarImage.text = [NSString stringWithFormat:@"总体评分：%@分",pf];

jionTimeLab.text = [NSString stringWithFormat:@"%@,已发布%@条货源", rootDict[@"jointime"], rootDict[@"fbcount"]];

telStr = rootDict[@"usertel"];

NSString \*priceStr = rootDict[@"price"];

priceLab.text = priceStr;

//是否关注

attentionStr = rootDict[@"isattention"];

if ([\_attentionStr isEqualToString:@"true"]) {

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_follow11"] forState:UIControlStateNormal];

} else {

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_unfollow11"] forState:UIControlStateNormal];

}

}

}

[\_myCustomerCarInfoTab reloadData];

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

[lbxManager showHudViewLabelText:@"获取订单信息失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

[lbxManager showHUDAction:1 showView:self.view];

}

- (UIButton \*)creeatbtnwithName:(NSString \*)name {

UIButton \*btn = [UIButton buttonWithType:UIButtonTypeCustom];

btn.layer.borderColor = [UIColor colorWithRed:34 / 255.0 green:97/ 255.0 blue:232 / 255.0 alpha:1].CGColor;

btn.layer.masksToBounds = YES;

btn.layer.cornerRadius = 5.0;

btn.layer.borderWidth = 1.0;

[btn setTitleColor:[UIColor whiteColor] forState:UIControlStateNormal];

btn.titleLabel.font = [UIFont systemFontOfSize:14];

btn.imageEdgeInsets = UIEdgeInsetsMake(5, 5, 5, 60);

btn.titleEdgeInsets = UIEdgeInsetsMake(5, -10, 5, 5);

[btn setTitle:name forState:UIControlStateNormal];

[btn setImage:[UIImage imageNamed:@"map"] forState:UIControlStateNormal];

[self.view addSubview:btn];

return btn;

}

- (void)setExtraCellLineHidden:(UITableView \*)tableView {

UIView \*view = [UIView new];

view.backgroundColor = [UIColor clearColor];

[tableView setTableFooterView:view];

}

#pragma mark-------label的宽度

-(CGFloat)creatLabelwideWithText:(NSString \*)textStr WithFont:(UIFont \*)sysFont{

UILabel \*label = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 1000, 30)];

UIFont \*font = sysFont;

[label setFont:font];

[label setNumberOfLines:0];

[label setBackgroundColor:[UIColor clearColor]];

NSString \*text = textStr;

CGSize size ;

NSDictionary \*dic = @{NSFontAttributeName:font};

size = [text boundingRectWithSize:CGSizeMake(99999.0, 30.0f) options:NSStringDrawingUsesLineFragmentOrigin|NSStringDrawingUsesFontLeading attributes:dic context:nil].size;

CGRect rect = label.frame;

rect.size = size;

//NSLog(@"==================================");

[label setFrame:rect];

[label setText:text];

return rect.size.width;

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}#import "MyCustomCarsInfoWithMapVC.h"

#import "UIImage+Rotate.h"

#import "MyCutomerCarsInfoViewController.h"

#import "ChatViewController.h"

#import "PublishProducInfotVC.h"

#define MYBUNDLE\_NAME @ "mapapi.bundle"

#define MYBUNDLE\_PATH [[[NSBundle mainBundle] resourcePath] stringByAppendingPathComponent: MYBUNDLE\_NAME]

#define MYBUNDLE [NSBundle bundleWithPath: MYBUNDLE\_PATH]

@interface CarsInfoRouteAnnotation : BMKPointAnnotation{

int \_type; ///<0:起点 1：终点 2：公交 3：地铁 4:驾乘 5:途经点

int \_degree;

}

@property (nonatomic) int type;

@property (nonatomic) int degree;

@end

@implementation CarsInfoRouteAnnotation

@synthesize type = \_type;

@synthesize degree = \_degree;

@end

@interface MyCustomCarsInfoWithMapVC ()<UITableViewDelegate, UITableViewDataSource, BMKMapViewDelegate, BMKRouteSearchDelegate, BMKGeoCodeSearchDelegate>

@property (nonatomic, strong) MBProgressHUD \*hud;

@property (nonatomic, strong)UITableView \*myTab;

@property (nonatomic, strong) UIView \*bigView;

@property (nonatomic, strong) UIView \*bottomView;

@property (nonatomic,retain)NSDictionary \* dataDic;

@property (nonatomic,retain)NSArray \* orderedKeys;

@property (nonatomic, strong) UILabel \*priceLab;

@property (nonatomic, strong) UILabel \*labelTitle;

@property (nonatomic, strong) UILabel \*StarImage;

@property (nonatomic, copy) NSString \*telStr;

@property (nonatomic, copy) NSString \*attentStr;

@property (nonatomic, copy) NSString \*attentionStr;

@property (nonatomic, strong) NSMutableDictionary \*myCarInfoDict;

@property (nonatomic, strong) UIButton \*mapBtn;

@property (nonatomic, copy) NSString \*startStr;

@property (nonatomic, copy) NSString \*arriveStr;

@property (nonatomic, strong) UIButton \*IMbtn;

@property (nonatomic, strong) UIButton \*phoneBtn;

@property (nonatomic, strong) UIButton \*attentionBtn;

@property (nonatomic, strong) UIView \*bg;

@property (nonatomic, strong) UIImageView \*myHeaderView;

@property (nonatomic, strong) UILabel \*usersTypeLab;

@property (nonatomic, strong)UILabel \*jionInTimeLab;//加入时间

@property (nonatomic, strong)UILabel \*driverNameLab;//姓名

@property (nonatomic, strong)UILabel \*realNameLab;//实名

@property (nonatomic, strong)UILabel \*typeLab;//类别

@property (nonatomic, assign)CGFloat driverNameLabWidth;

@property (nonatomic, copy) NSString \*myNameStr;

@property (nonatomic, strong) UILabel \*driverStarImage;

@property (nonatomic, copy) NSString \*caddress;

@property (nonatomic, copy) NSString \*daddress;

@property (nonatomic, copy) NSString \*distance1;

@property (nonatomic, strong) UILabel \*distanceLab;

@property (nonatomic, strong)BMKMapView \*mapView;

@property(nonatomic, strong) BMKUserLocation \*userStartLocation;//用户的起始位置

@property(nonatomic, strong) BMKUserLocation \*userEndLocation; // 用户的目的位置

@property (nonatomic, strong) BMKGeoCodeSearch \*geocodesearch; //地理编码

@property (nonatomic, strong) BMKDrivingRouteLine \*plan;

@property (nonatomic, strong) BMKRouteSearch \*routesearch; //驾车路线规划

@property (nonatomic, assign) BOOL isGeoSearch;

@property (nonatomic, strong) BMKPointAnnotation \*item;

@property (nonatomic, assign)double longitude;

@property (nonatomic, assign)double Latitude;

@property (nonatomic, assign)double longitude1;

@property (nonatomic, assign)double Latitude1;

@property (nonatomic, copy) NSString \*carInfoStr;

@property (nonatomic, strong) UIView \*smallBg;

@end

@implementation MyCustomCarsInfoWithMapVC

- (void)viewWillAppear:(BOOL)animated {

[super viewWillAppear:animated];

\_mapView.delegate = self; // 此处记得不用的时候需要置nil，否则影响内存的释放

\_routesearch.delegate = self; // 此处记得不用的时候需要置nil，否则影响内存的释放

\_geocodesearch.delegate = self;

//请求数据

[self getCarsSourceInfoData];

}

-(void)viewWillDisappear:(BOOL)animated {

[\_mapView viewWillDisappear];

\_mapView.delegate = nil; // 不用时，置nil

\_routesearch.delegate = nil; // 不用时，置nil

\_geocodesearch.delegate = nil; // 不用时，置nil

}

- (NSString\*)getMyBundlePath1:(NSString \*)filename

{

NSBundle \* libBundle = MYBUNDLE ;

if ( libBundle && filename ){

NSString \* s=[[libBundle resourcePath ] stringByAppendingPathComponent:filename];

return s;

}

return nil ;

}

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.title = @"车源详情";

\_myTab = [[UITableView alloc] initWithFrame:CGRectMake(0, 70, kDeviceWidth, 110) style:UITableViewStylePlain];

\_myTab.delegate = self;

\_myTab.dataSource = self;

[self.view addSubview:\_myTab];

\_mapView = [[BMKMapView alloc] initWithFrame:CGRectMake(0, 100, kDeviceWidth, kDeviceHeight - 100 - 64 - 50)];

\_mapView.buildingsEnabled = YES;

\_mapView.overlookEnabled = YES;

\_mapView.showMapScaleBar = YES;

\_mapView.overlooking = -45;

[self.view addSubview:\_mapView];

//地理 编码

\_geocodesearch = [[BMKGeoCodeSearch alloc] init];

//驾车路线规划

\_routesearch = [[BMKRouteSearch alloc]init];

[self initHeaderView];

\_distanceLab = [[UILabel alloc] init];

\_distanceLab.frame = CGRectMake(kDeviceWidth - 110, 10, 100, 30);

//\_distanceLab.backgroundColor = [UIColor redColor];

\_distanceLab.font = [UIFont systemFontOfSize:13];

\_distanceLab.textColor = [UIColor blackColor];

\_distanceLab.textAlignment = NSTextAlignmentRight;

[\_mapView addSubview:\_distanceLab];

\_bottomView = [[UIView alloc] init];

\_bottomView.frame = CGRectMake(0, kDeviceHeight - 110, kDeviceWidth, 50);

\_bottomView.backgroundColor = [UIColor whiteColor];

[self.view addSubview:\_bottomView];

[self initBottomView];

}

#pragma mark------ UIView

- (void)initHeaderView {

\_bigView = [[UIView alloc] init];

\_bigView.frame = CGRectMake(0, 0, kDeviceWidth, 70);

[self.view addSubview:\_bigView];

\_myHeaderView = [[UIImageView alloc] init];

\_myHeaderView.frame = CGRectMake(10, 5, 60, 60);

\_myHeaderView.layer.masksToBounds = YES;

\_myHeaderView.layer.cornerRadius = 30;

\_myHeaderView.image = [UIImage imageNamed:@"default\_pic"];

[\_bigView addSubview:\_myHeaderView];

\_driverNameLab = [[UILabel alloc] init];

\_driverNameLab.font = [UIFont systemFontOfSize:13];

[\_bigView addSubview:\_driverNameLab];

\_realNameLab = [[UILabel alloc] init];

\_realNameLab.layer.masksToBounds = YES;

\_realNameLab.text = @"实";

\_realNameLab.textAlignment = NSTextAlignmentCenter;

\_realNameLab.textColor = [UIColor whiteColor];

\_realNameLab.font = [UIFont systemFontOfSize:13];

\_realNameLab.layer.cornerRadius = 2;

\_realNameLab.backgroundColor = [UIColor colorWithRed:250 / 255.0 green:163 / 255.0 blue:26 / 255.0 alpha:1];

[\_bigView addSubview:\_realNameLab];

\_typeLab = [[UILabel alloc] init];

\_typeLab.text = @"车";

\_typeLab.textAlignment = NSTextAlignmentCenter;

\_typeLab.textColor = [UIColor whiteColor];

\_typeLab.font = [UIFont systemFontOfSize:13];

\_typeLab.layer.masksToBounds = YES;

\_typeLab.layer.cornerRadius = 2;

\_typeLab.backgroundColor = [UIColor colorWithRed:24 / 255.0 green:205 / 255.0 blue:186 / 255.0 alpha:1];

[\_bigView addSubview:\_typeLab];

//星星

UIView \*startView = [[UIView alloc] init];

startView.frame = CGRectMake(80, 30, 250, 20);

[\_bigView addSubview:startView];

\_driverStarImage = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 250, 20)];

\_driverStarImage.font = [UIFont systemFontOfSize:13];

[startView addSubview:\_driverStarImage];

//加入时间

\_jionInTimeLab = [[UILabel alloc] init];

\_jionInTimeLab.frame = CGRectMake(80, 50, 200, 20);

\_jionInTimeLab.font = [UIFont systemFontOfSize:12];

[\_bigView addSubview:\_jionInTimeLab];

\_attentionBtn = [UIButton buttonWithType:UIButtonTypeCustom];

\_attentionBtn.frame = CGRectMake(kDeviceWidth - 100, 25, 25, 25);

\_attentionBtn.layer.masksToBounds = YES;

[\_attentionBtn setImageEdgeInsets:UIEdgeInsetsMake(6, 3, 0, 3)];

\_attentionBtn.imageView.contentMode = UIViewContentModeScaleAspectFit;

[\_attentionBtn addTarget:self action:@selector(attenBtnAction:) forControlEvents:UIControlEventTouchUpInside];

[\_bigView addSubview:\_attentionBtn];

\_IMbtn = [UIButton buttonWithType:UIButtonTypeCustom];

\_IMbtn.frame = CGRectMake(kDeviceWidth - 70, 25, 25, 25);

[\_IMbtn setImageEdgeInsets:UIEdgeInsetsMake(6, 3, 0, 3)];

\_IMbtn.imageView.contentMode = UIViewContentModeScaleAspectFit;

[\_IMbtn setImage:[UIImage imageNamed:@"IMBtn"] forState:UIControlStateNormal];

[\_IMbtn addTarget:self action:@selector(IMAction:) forControlEvents:UIControlEventTouchUpInside];

[\_bigView addSubview:\_IMbtn];

NSString \*shType = [[NSUserDefaults standardUserDefaults]stringForKey:@"sh"];

if ([shType isEqualToString:@"已审核"]) {

phoneBtn = [UIButton buttonWithType:UIButtonTypeCustom];

phoneBtn.frame = CGRectMake(kDeviceWidth - 40, 25, 25, 25);

[\_phoneBtn setImageEdgeInsets:UIEdgeInsetsMake(6, 3, 0, 3)];

phoneBtn.layer.masksToBounds = YES;

phoneBtn.layer.cornerRadius = 12.5;

[\_phoneBtn setImage:[UIImage imageNamed:@"cellBtn11"] forState:UIControlStateNormal];

[\_phoneBtn addTarget:self action:@selector(phoneBtnAction:) forControlEvents:UIControlEventTouchUpInside];

[\_bigView addSubview:\_phoneBtn];

}

#pragma mark---- UITableViewDataSource

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section {

return 1;

}

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath {

return 30;

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

static NSString \* identifer = @"reuse";

UITableViewCell \* cell = [tableView cellForRowAtIndexPath:indexPath];

if (!cell) {

cell = [[UITableViewCell alloc]initWithStyle:UITableViewCellStyleValue1 reuseIdentifier:identifer];

}

cell.accessoryType = UITableViewCellAccessoryDisclosureIndicator; //显示最右边的箭头

cell.selectionStyle = UITableViewCellSelectionStyleNone;

cell.textLabel.text = @"车源信息";

cell.textLabel.font = [UIFont systemFontOfSize:13];

cell.detailTextLabel.font = [UIFont systemFontOfSize:13];

cell.detailTextLabel.textColor = [UIColor blackColor];

cell.detailTextLabel.text = \_carInfoStr;

return cell;

}

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath {

MyCutomerCarsInfoViewController \*carInfoVC = [[MyCutomerCarsInfoViewController alloc] init];

carInfoVC.carID = \_carID;

carInfoVC.phoneStr = \_phoneStr;

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

}

#pragma amrk------数据

//关注

- (void)setAttentionWithData {

NSLog(@"已经关注");

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSMutableDictionary \*attentionDict = [[NSMutableDictionary alloc] init];

[attentionDict setValue:\_phoneStr forKey:@"Duser"];

[attentionDict setValue:dict[@"username"] forKey:@"username"];

NSLog(@"已经关注attentionDict:%@", attentionDict);

[BJHttpTool Post:KAttDriver parameters:attentionDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*attentionData = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"attentionData = %@",attentionData);

if ([attentionData[@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"已成功关注" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_follow11"] forState:UIControlStateNormal];

}

if ([[attentionData objectForKey:@"errcode"] intValue] == 1) {

[self cancleAttentionWithData];

NSLog(@"跑到了这里！！！！");

}

}

} failure:^(NSError \*error) {

NSLog(@"error = %@", error);

[lbxManager showHudViewLabelText:@"关注失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

}

//取消关注

- (void)cancleAttentionWithData {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSMutableDictionary \*cancleAttentionDict = [[NSMutableDictionary alloc] init];

[cancleAttentionDict setValue:dict[@"username"] forKey:@"username"];

[cancleAttentionDict setValue:\_phoneStr forKey:@"Duser"];

NSLog(@"取消关注cancleAttentionDict:%@", cancleAttentionDict);

[BJHttpTool Post:KCancelAttDriver parameters:cancleAttentionDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*cancleAttentionData = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"cancleAttentionData = %@",cancleAttentionData);

if ([[cancleAttentionData objectForKey:@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"取消关注" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_unfollow11"] forState:UIControlStateNormal];

attentStr = @"取消关注成功";

}

}

} failure:^(NSError \*error) {

NSLog(@"error = %@", error);

[lbxManager showHudViewLabelText:@"取消关注失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

}

//车源详情

- (void)getCarsSourceInfoData {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

//[lbxManager showHUDAction:0 showView:self.view];

NSMutableDictionary \*paramDict = [[NSMutableDictionary alloc] init];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

[paramDict setValue:dict[@"username"] forKey:@"username"];

[paramDict setValue:\_carID forKey:@"tid"];

[paramDict setValue:\_phoneStr forKey:@"czuser"];

NSLog(@"所有的参数 = %@", paramDict);

[BJHttpTool Post:KGetcarfbInfoByUser parameters:paramDict success:^(id responseObject) {

if (responseObject) {

NSError \*error = nil;

NSMutableDictionary \*rootDict = [[NSMutableDictionary alloc] init];

rootDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"车源详情rootDict = %@",rootDict);

if ([[rootDict objectForKey:@"errcode"] intValue] == 0) {

//[lbxManager showHudViewLabelText:@"获取车源信息成功" detailsLabelText:nil afterDelay:kAfterDelayTime];

//[lbxManager showHUDAction:1 showView:self.view];

self.myCarInfoDict = [NSMutableDictionary dictionaryWithDictionary:rootDict];

NSURL \*url = [NSURL URLWithString:rootDict[@"headimg"]];

if (url == nil) {

myHeaderView.image = [UIImage imageNamed:@"default\_pic"];

}

else{

[\_myHeaderView sd\_setImageWithURL:url placeholderImage:[UIImage imageNamed:@"default\_pic"]];

}

myNameStr = rootDict[@"name"];

//姓名的宽度

driverNameLabWidth = [self creatLabelwideWithText:\_myNameStr WithFont:[UIFont systemFontOfSize:14]];

driverNameLab.text = \_myNameStr;

driverNameLab.frame = CGRectMake(80, 10, \_driverNameLabWidth, 20);

realNameLab.frame = CGRectMake(\_driverNameLabWidth + 90, 10, 20, 20);

typeLab.frame = CGRectMake(\_driverNameLabWidth + 113, 10, 20, 20);

//NSString \*str1 = rootDict[@"fb\_type"];

//if ([str1 isEqualToString:@"公司"]) {

// usersTypeLab.text = @"企";

//} else {

// usersTypeLab.text = @"个";

//}

carInfoStr = [NSString stringWithFormat:@"%@/%@/%@", rootDict[@"cph"], rootDict[@"csize"], rootDict[@"cartype"]];

NSString \*pf = rootDict[@"pf"];

StarImage.text = [NSString stringWithFormat:@"总体评分：%@分",pf];

jionInTimeLab.text = [NSString stringWithFormat:@"%@加入,已发布%@条货源", rootDict[@"jointime"], rootDict[@"fbcount"]];

telStr = rootDict[@"usertel"];

NSString \*priceStr = rootDict[@"price"];

priceLab.text = priceStr;

caddress = [NSString stringWithFormat:@"%@%@%@%@", rootDict[@"cprovince"], rootDict[@"ccity"], rootDict[@"carea"], rootDict[@"caddress"]];

daddress = [NSString stringWithFormat:@"%@%@%@%@", rootDict[@"dprovince"], rootDict[@"dcity"], rootDict[@"darea"], rootDict[@"daddress"]];

if (![\_daddress isEqualToString:@""] || ![\_caddress isEqualToString:@""]) {

NSLog(@"\_caddress = %@, \_daddress = %@", \_caddress, \_daddress);

[self getLatitudeAndLongitude];

}

//是否关注

attentionStr = rootDict[@"isattention"];

if ([\_attentionStr isEqualToString:@"true"]) {

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_follow11"] forState:UIControlStateNormal];

} else {

[\_attentionBtn setImage:[UIImage imageNamed:@"order\_unfollow11"] forState:UIControlStateNormal];

}

}

}

[\_myTab reloadData];

} failure:^(NSError \*error) {

NSLog(@"error = %@",error.description);

//[lbxManager showHudViewLabelText:@"获取订单信息失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

//[lbxManager showHUDAction:1 showView:self.view];

}];

//[lbxManager showHUDAction:1 showView:self.view];

}

#pragma mark----- 路径规划

- (void)getLatitudeAndLongitude {

BMKGeoCodeSearchOption \*geocodeSearchOption = [[BMKGeoCodeSearchOption alloc]init];

if (\_isGeoSearch == false) {

geocodeSearchOption.address = \_caddress;

} else if (\_isGeoSearch == true) {

geocodeSearchOption.address = \_daddress;

}

BOOL flag3 = [\_geocodesearch geoCode:geocodeSearchOption];

if(flag3)

{

NSLog(@"geo检索发送成功");

}

else

{

NSLog(@"geo检索发送失败");

}

}

- (void)onGetGeoCodeResult:(BMKGeoCodeSearch \*)searcher result:(BMKGeoCodeResult \*)result errorCode:(BMKSearchErrorCode)error {

if (error == 0) {

item = [[BMKPointAnnotation alloc]init];

item.coordinate = result.location;

}

if (\_isGeoSearch == false) {

longitude = \_item.coordinate.longitude;

Latitude = \_item.coordinate.latitude;

NSLog(@"出发的经度:%f,出发纬度:%f" ,\_Latitude ,\_longitude);

isGeoSearch = true;

[self getLatitudeAndLongitude];

} else if (\_isGeoSearch == true) {

longitude1 = \_item.coordinate.longitude;

Latitude1 = \_item.coordinate.latitude;

NSLog(@"到达的经度:%f,到达纬度:%f" ,\_Latitude1 ,\_longitude1);

}

//获得经纬度 后进行 路径规划的检索

[self getDriverPathWithAddress];

}

- (void)getDriverPathWithAddress {

BMKPlanNode\* start = [[BMKPlanNode alloc]init];

start.pt = CLLocationCoordinate2DMake(\_Latitude, \_longitude);

BMKPlanNode\* end = [[BMKPlanNode alloc]init];

end.pt = CLLocationCoordinate2DMake(\_Latitude1, \_longitude1);

BMKDrivingRoutePlanOption \*drivingRouteSearchOption = [[BMKDrivingRoutePlanOption alloc]init];

drivingRouteSearchOption.from = start;

drivingRouteSearchOption.to = end;

BOOL flag77 = [\_routesearch drivingSearch:drivingRouteSearchOption];

if(flag77)

{

NSLog(@"car检索发送成功");

if ([\_distance1 isEqualToString:@""] || \_distance1.length == 0) {

[self showWaitWithMessage:@"正在加载中..."];

}

}

else

{

NSLog(@"car检索发送失败");

}

}

- (void)onGetDrivingRouteResult:(BMKRouteSearch\*)searcher result:(BMKDrivingRouteResult\*)result errorCode:(BMKSearchErrorCode)error

{

NSArray\* array = [NSArray arrayWithArray:\_mapView.annotations];

[\_mapView removeAnnotations:array];

array = [NSArray arrayWithArray:\_mapView.overlays];

[\_mapView removeOverlays:array];

if (error == BMK\_SEARCH\_NO\_ERROR) {

//处理正常结果

[self closeWaitView];

NSLog(@"正常结果%@", result.routes);

plan = (BMKDrivingRouteLine\*)[result.routes objectAtIndex:0];

BMKRouteLine \*line = \_plan;

int distance = line.distance / 1000;

distance1 = [NSString stringWithFormat:@"%d", distance];

NSLog(@"两地的距离%@", \_distance1);

distanceLab.text = [NSString stringWithFormat:@"%@公里", \_distance1];

// 计算路线方案中的路段数目

NSInteger size = [\_plan.steps count];

int planPointCounts = 0;

for (int i = 0; i < size; i++) {

BMKDrivingStep\* transitStep = [\_plan.steps objectAtIndex:i];

if (i == 0) {

CarsInfoRouteAnnotation\* item = [[CarsInfoRouteAnnotation alloc]init];

item.coordinate = \_plan.starting.location;

item.title = @"起点";

item.type = 0;

[\_mapView addAnnotation:item];

} else if (i == size - 1) {

CarsInfoRouteAnnotation\* item = [[CarsInfoRouteAnnotation alloc]init];

item.coordinate = \_plan.terminal.location;

item.title = @"终点";

item.type = 1;

[\_mapView addAnnotation:item]; // 添加起点标注

}

//添加 节点

CarsInfoRouteAnnotation \*Item = [[CarsInfoRouteAnnotation alloc] init];

Item.coordinate = transitStep.entrace.location;

Item.title = transitStep.entraceInstruction;

Item.degree = transitStep.direction \* 30;

Item.type = 4;

[\_mapView addAnnotation:Item];

//③轨迹点总数累计

planPointCounts += transitStep.pointsCount;

}

//④添加 途经点

if (\_plan.wayPoints) {

for (BMKPlanNode \*tempNode in \_plan.wayPoints) {

CarsInfoRouteAnnotation\* item = [[CarsInfoRouteAnnotation alloc]init];

item = [[CarsInfoRouteAnnotation alloc]init];

item.coordinate = tempNode.pt;

item.type = 5;

item.title = tempNode.name;

[\_mapView addAnnotation:item];

}

}

//⑤轨迹点

BMKMapPoint \*tempPoints = new BMKMapPoint[planPointCounts];

int i = 0;

for (int j = 0; j < size; j++) {

BMKTransitStep \*transitStep = [\_plan.steps objectAtIndex:j];

int k=0;

for(k=0;k<transitStep.pointsCount;k++) {

tempPoints[i].x = transitStep.points[k].x;

tempPoints[i].y = transitStep.points[k].y;

i++;

}

}

//⑥通过points构建 BMKPolyline

BMKPolyline \*polyLine = [BMKPolyline polylineWithPoints:tempPoints count:planPointCounts];

[\_mapView addOverlay:polyLine];

delete [] tempPoints;

[self mapViewFitPolyLine:polyLine];

}

}

//根据polyline设置地图范围

- (void)mapViewFitPolyLine:(BMKPolyline \*) polyLine {

CGFloat ltX, ltY, rbX, rbY;

if (polyLine.pointCount < 1) {

return;

}

BMKMapPoint pt = polyLine.points[0];

ltX = pt.x, ltY = pt.y;

rbX = pt.x, rbY = pt.y;

for (int i = 1; i < polyLine.pointCount; i++) {

BMKMapPoint pt = polyLine.points[i];

if (pt.x < ltX) {

ltX = pt.x;

}

if (pt.x > rbX) {

rbX = pt.x;

}

if (pt.y > ltY) {

ltY = pt.y;

}

if (pt.y < rbY) {

rbY = pt.y;

}

}

BMKMapRect rect;

rect.origin = BMKMapPointMake(ltX , ltY);

rect.size = BMKMapSizeMake(rbX - ltX, rbY - ltY);

[\_mapView setVisibleMapRect:rect];

\_mapView.zoomLevel = \_mapView.zoomLevel - 0.3;

}

#pragma mark - mapViewDelegate

- (BMKOverlayView \*)mapView:(BMKMapView \*)mapView viewForOverlay:(id<BMKOverlay>)overlay

{

if ([overlay isKindOfClass:[BMKPolyline class]]) {

BMKPolylineView \*polylineView = [[BMKPolylineView alloc] initWithOverlay:overlay];

polylineView.strokeColor = RGBACOLOR(194, 23, 239, 1);

polylineView.lineWidth = 3.0;

return polylineView;

}

return nil;

}

#pragma mark-------------------------///////

#pragma mark -显示大头针

- (BMKAnnotationView \*)mapView:(BMKMapView \*)mapView viewForAnnotation:(id <BMKAnnotation>)annotation{

if (![annotation isKindOfClass:[CarsInfoRouteAnnotation class]]) return nil;

return [self getRouteAnnotationView:mapView viewForAnnotation:(CarsInfoRouteAnnotation \*)annotation];

}

#pragma mark 获取路线的标注，显示到地图

- (BMKAnnotationView\*)getRouteAnnotationView:(BMKMapView \*)mapview viewForAnnotation:(CarsInfoRouteAnnotation\*)routeAnnotation{

BMKAnnotationView \*view = nil;

switch (routeAnnotation.type) {

case 0:

{

view = [mapview dequeueReusableAnnotationViewWithIdentifier:@"start\_node"];

if (view == nil) {

view = [[BMKAnnotationView alloc] initWithAnnotation:routeAnnotation reuseIdentifier:@"start\_node"];

view.image = [UIImage imageNamed:@"b\_start(1)"];

view.size = CGSizeMake(24, 24);

view.canShowCallout = true;

}

view.annotation = routeAnnotation;

}

break;

case 1:

{

view = [mapview dequeueReusableAnnotationViewWithIdentifier:@"end\_node"];

if (view == nil) {

view = [[BMKAnnotationView alloc] initWithAnnotation:routeAnnotation reuseIdentifier:@"end\_node"];

view.image = [UIImage imageNamed:@"b\_end(1)"];

view.size = CGSizeMake(24, 24);

view.canShowCallout = true;

}

view.annotation =routeAnnotation;

}

break;

case 4:

{

view = [mapview dequeueReusableAnnotationViewWithIdentifier:@"route\_node"];

if (view == nil) {

view = [[BMKAnnotationView alloc] initWithAnnotation:routeAnnotation reuseIdentifier:@"route\_node"];

view.canShowCallout = true;

} else {

[view setNeedsDisplay];

}

UIImage \*image = [UIImage imageWithContentsOfFile:[self getMyBundlePath1:@"images/icon\_direction"]];

view.image = [image imageRotatedByDegrees:routeAnnotation.degree];

view.annotation = routeAnnotation;

}

break;

default:

break;

}

return view;

}

- (void)initBottomView {

UILabel \*oneLab = [[UILabel alloc] init];

oneLab.frame = CGRectMake(20, 10, 70, 30);

oneLab.text = @"运输价格:";

oneLab.font = [UIFont systemFontOfSize:15];

[\_bottomView addSubview:oneLab];

\_priceLab = [[UILabel alloc] init];

\_priceLab.frame = CGRectMake(90, 10, kDeviceWidth / 2 - 82, 30);

\_priceLab.text = @"100元";

\_priceLab.textColor = [UIColor colorWithRed:230/255.0 green:71/255.0 blue:37/255.0 alpha:1];

[\_bottomView addSubview:\_priceLab];

UIButton \*competeOrderBtn = [UIButton buttonWithType:UIButtonTypeCustom];

competeOrderBtn.frame = CGRectMake(kDeviceWidth / 2 + 30, 10, kDeviceWidth / 2 - 40, 30);

competeOrderBtn.layer.masksToBounds = YES;

competeOrderBtn.layer.cornerRadius = 5;

competeOrderBtn.backgroundColor = [UIColor colorWithRed:230/255.0 green:71/255.0 blue:37/255.0 alpha:1];

[competeOrderBtn setTitle:@"约车" forState:UIControlStateNormal];

[competeOrderBtn addTarget:self action:@selector(competeOrderAction:) forControlEvents:UIControlEventTouchUpInside];

[\_bottomView addSubview:competeOrderBtn];

}

- (void)competeOrderAction:(id) sender1 {

NSLog(@"约车发货！！！！！");

[self creatUI];

}

- (void)creatUI {

\_smallBg = [[UIView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight)];

\_smallBg.backgroundColor = [UIColor colorWithRed:0 green:0 blue:0 alpha:0.5];

CAShapeLayer \*maskLayer=[[CAShapeLayer alloc] init];

CGRect maskRect=CGRectMake(0,0,kDeviceWidth,kDeviceHeight); //蒙版所有遮住的W和H

CGMutablePathRef path=CGPathCreateMutable();

CGPathAddRect(path, nil, maskRect);

[maskLayer setPath:path];

CGPathRelease(path);

\_smallBg.layer.mask = maskLayer;

[self.view addSubview:\_smallBg];

UIView \*showView=[[UIView alloc]initWithFrame:CGRectMake(leftDistance, (kDeviceHeight - 160) / 2 - (kDeviceHeight - 160)/4/2, kDeviceWidth-leftDistance\*2, 175)];

[\_smallBg addSubview:showView];

[showView setBackgroundColor:[UIColor whiteColor]];

showView.layer.cornerRadius=5.0;

showView.layer.masksToBounds=YES;

UILabel \*labelTitle = [[UILabel alloc] init];

labelTitle.frame = CGRectMake(0, 0, kDeviceWidth-leftDistance\*2, 30);

labelTitle.text = @"好运道提示";

labelTitle.font=[UIFont systemFontOfSize:17];

labelTitle.textAlignment=NSTextAlignmentCenter;

[showView addSubview:labelTitle];

UILabel \*contentLab = [[UILabel alloc] init];

contentLab.frame = CGRectMake(10, 35, kDeviceWidth-leftDistance\*2 - 20, 80);

contentLab.text = @"好运道提醒您注意，即使和车主达成运输协议，请仍在本系统下单，以便我们提供物流网上追踪，支付保障，优质评价等来维护您的正当权利.";

contentLab.font = [UIFont systemFontOfSize:15];

contentLab.numberOfLines = 0;

contentLab.lineBreakMode = NSLineBreakByWordWrapping;

[showView addSubview:contentLab];

UIButton \*affirmBtn = [UIButton buttonWithType:UIButtonTypeCustom];

affirmBtn.frame = CGRectMake((kDeviceWidth - 80 - leftDistance\*2) / 2, 135, 80, 30);

affirmBtn.layer.borderWidth = 0.5;

affirmBtn.layer.borderColor = kOrangeColor.CGColor;

affirmBtn.layer.masksToBounds = YES;

affirmBtn.layer.cornerRadius = 5;

[affirmBtn setTitle:@"确认" forState:UIControlStateNormal];

[affirmBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

[affirmBtn addTarget:self action:@selector(affirmAction:) forControlEvents:UIControlEventTouchUpInside];

[showView addSubview:affirmBtn];

}

- (void)affirmAction:(UIButton \*)sender {

[\_smallBg removeFromSuperview];

PublishProducInfotVC \*myPublishVC = [[PublishProducInfotVC alloc] init];

myPublishVC.typeStr = @"约车";

myPublishVC.myCarsInfoDict = self.myCarInfoDict;

myPublishVC.orderID = @"";

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

}

//关注

- (void)attenBtnAction:(UIButton \*)attenSender {

if ([\_attentStr isEqualToString:@"取消关注成功"] ||![\_attentionStr isEqualToString:@"True"]) {

[self setAttentionWithData];

} else {

[self cancleAttentionWithData];

}

}

//IM

- (void)IMAction:(UIButton \*)sender {

NSLog(@"IM ");

ChatViewController \*chatVC = [[ChatViewController alloc] initWithConversationChatter:self.phoneStr conversationType:EMConversationTypeChat];

chatVC.title = @"对话信息";

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

}

//电话

- (void)phoneBtnAction:(UIButton \*)sender1 {

NSLog(@"电话!!!! ");

[self openURLToCall];

}

#pragma mark------拨打电话

- (void)openURLToCall {

NSLog(@"货主电话");

NSString \*phoneStr = self.phoneStr;

//防止电话号码 为空时程序崩溃

if ([self.phoneStr isEqual: @""] || self.phoneStr == nil || self.phoneStr.length == 0) {

return ;

} else {

UIAlertView \*alertView = [[UIAlertView alloc] initWithTitle:@"联系车主" message:phoneStr delegate:self cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

[alertView show];

}

}

#pragma mark - UIAlertView的代理

- (void)alertView:(UIAlertView \*)alertView clickedButtonAtIndex:(NSInteger)buttonIndex

{

if (buttonIndex == alertView.cancelButtonIndex) {

NSLog(@"取消");

} else if (buttonIndex == 1){

[self openPhone];

}

}

//打客服电话

- (void)openPhone {

NSMutableString \*str = [[NSMutableString alloc] initWithFormat:@"tel:%@",self.phoneStr];

UIWebView \* callWebview = [[UIWebView alloc] init];

[callWebview loadRequest:[NSURLRequest requestWithURL:[NSURL URLWithString:str]]];

[self.view addSubview:callWebview];

}

-(void)showWaitWithMessage:(NSString \*)message

{

if (self.hud == nil) {

hud = [[MBProgressHUD alloc] initWithView:self.view];

self.hud.labelText = message;

self.hud.yOffset = -30;

[self.view addSubview:self.hud];

[self.hud show:YES];

}

}

-(void)closeWaitView

{

if (self.hud) {

[self.hud removeFromSuperview];

self.hud = nil;

}

}

#pragma mark-------label的宽度

-(CGFloat)creatLabelwideWithText:(NSString \*)textStr WithFont:(UIFont \*)sysFont{

UILabel \*label = [[UILabel alloc] initWithFrame:CGRectMake(0, 0, 1000, 30)];

UIFont \*font = sysFont;

[label setFont:font];

[label setNumberOfLines:0];

[label setBackgroundColor:[UIColor clearColor]];

NSString \*text = textStr;

CGSize size ;

NSDictionary \*dic = @{NSFontAttributeName:font};

size = [text boundingRectWithSize:CGSizeMake(99999.0, 30.0f) options:NSStringDrawingUsesLineFragmentOrigin|NSStringDrawingUsesFontLeading attributes:dic context:nil].size;

CGRect rect = label.frame;

rect.size = size;

//NSLog(@"==================================");

[label setFrame:rect];

[label setText:text];

return rect.size.width;

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}

#import "BJMyNearbyBussinessListViewController.h"

#import "BJMyNearbyBusinessViewController.h"//地图列表

#import "HYDGoogListCell.h"//列表

#import "MyCustomCarsInfoWithMapVC.h"//车源详情（地图页面）

#import "BJGoodsDetailViewController.h"//货源详情

#import "LZRelayoutButton.h"

#import "HomeChildController.h"

#import "BJLocation.h"

@interface BJMyNearbyBussinessListViewController ()<UITableViewDelegate, UITableViewDataSource, UICollectionViewDelegate,UICollectionViewDataSource>

@property (nonatomic, strong) UITableView \*myOrderList;

@property (nonatomic, strong)UILabel \*sorryLabel;

@property (nonatomic, assign) NSInteger Type; // 1:始发地 2:目的地

@property (nonatomic, strong) UIButton \*choseTypeBtn;

@property (nonatomic, strong) UIView \*smallView;

@property (nonatomic, assign) BOOL isChoseExist;

@property (nonatomic, strong) NSMutableDictionary \*paramDict;

@property (nonatomic, strong) UIButton \*startBtn;

@property (nonatomic, strong) UILabel \*startLab;

@property (nonatomic, strong) UIButton \*arriveBtn;

@property (nonatomic, strong) UILabel \*arriveLab;

@property (strong, nonatomic) NSArray \*provinces;

@property (strong, nonatomic) NSArray \*cities;

@property (strong, nonatomic) NSArray \*areas;

@property (strong, nonatomic) BJLocation \*locate;

@property (nonatomic, strong) UIView \*selectAddressView; // 选择地点视图

@property (nonatomic, strong) UIButton \*selectAllBtn;

@property (nonatomic, strong) UIButton \*gobackBtn;

@property (nonatomic, strong) UIButton \*returnBtn;

@property (nonatomic, strong) UICollectionView \*collectionView;

@property (nonatomic, assign) NSInteger level; // 1:省级 2:市级

@property (nonatomic, strong) NSString \*choseStr;

@property (nonatomic, copy)NSString \*value1;

@property (nonatomic, copy)NSString \*value2;

@property (nonatomic, assign) int page1;

@property (nonatomic, assign) NSInteger allCount;

@property (nonatomic, strong) NSMutableArray \*dataArr;

@end

@implementation BJMyNearbyBussinessListViewController

- (void)readDataFormPlist {

NSString \* filePath = [[NSBundle mainBundle]pathForResource:@"area" ofType:@"plist"];

self.provinces = [@[] mutableCopy];

self.provinces = [NSArray arrayWithContentsOfFile:filePath];

\_cities = [[\_provinces objectAtIndex:0] objectForKey:@"cities"];

self.locate.state = [[\_provinces objectAtIndex:0] objectForKey:@"state"];

self.locate.city = [[\_cities objectAtIndex:0] objectForKey:@"city"];

\_areas = [[\_cities objectAtIndex:0] objectForKey:@"areas"];

if (\_areas.count > 0) {

self.locate.district = [\_areas objectAtIndex:0];

} else{

self.locate.district = @"";

}

}

- (void)viewWillAppear:(BOOL)animated{

[super viewWillAppear:YES];

[\_sorryLabel setHidden:YES];

}

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.title = @"搜索车源";

[self readDataFormPlist];//plist文件

\_paramDict = [[NSMutableDictionary alloc] init];

\_dataArr = [@[] mutableCopy];

[\_paramDict setValue:@"离我最近" forKey:@"rank"];

[\_paramDict setValue:@"附近车源" forKey:@"type"];

NSString \*lat = [NSString stringWithFormat:@"%f", \_lat];

NSString \*lng = [NSString stringWithFormat:@"%f", \_lng];

[\_paramDict setValue:lat forKey:@"x"]; //经度

[\_paramDict setValue:lng forKey:@"y"]; //纬度

[\_paramDict setValue:@"" forKey:@"cprovince"];

[\_paramDict setValue:@"" forKey:@"ccity"];

[\_paramDict setValue:@"" forKey:@"dprovince"];

[\_paramDict setValue:@"" forKey:@"dcity"];

\_isChoseExist = false;

[self initHeaderView];

//0 表示没有筛选的条件

\_Type = 0;

\_myOrderList = [[UITableView alloc] initWithFrame:CGRectMake(0, 70, kDeviceWidth, kDeviceHeight -70-40-64) style:UITableViewStylePlain];

\_myOrderList.delegate = self;

\_myOrderList.dataSource = self;

[self.view addSubview:\_myOrderList];

\_myOrderList.tableFooterView = [[UIView alloc]init];

[\_myOrderList setSeparatorInset:UIEdgeInsetsMake(0, 0, 0, 0)];

[self setExtraCellLineHidden:\_myOrderList];

\_sorryLabel = [[UILabel alloc]initWithFrame:CGRectMake(0, self.view.frame.size.height/2.0-50, kDeviceWidth, 30)];

\_sorryLabel.text = @"暂无车源";

\_sorryLabel.textAlignment = NSTextAlignmentCenter;

\_sorryLabel.textColor = [UIColor blackColor];

\_sorryLabel.font = [UIFont systemFontOfSize:17];

[self.view addSubview:\_sorryLabel];

[\_sorryLabel setHidden:YES];

[self initBootomView];

//筛选按钮

UIBarButtonItem \*searchItem = [[UIBarButtonItem alloc] initWithTitle:@"筛选" style:UIBarButtonItemStylePlain target:self action:@selector(searchClick)];

searchItem.tintColor = [UIColor whiteColor];

self.navigationItem.rightBarButtonItem = searchItem;

\_page1 = 0;

[self getDataWithNum:\_page1];

\_myOrderList.header = [MJRefreshNormalHeader headerWithRefreshingBlock:^{

[self pullDownRereshing];

}];

\_myOrderList.footer = [MJRefreshBackNormalFooter footerWithRefreshingBlock:^{

[self getmoreData];

}];

}

//下拉刷新数据

- (void)pullDownRereshing {

\_page1 = 0;

[\_dataArr removeAllObjects];

[self getDataWithNum:\_page1];

}

//上拉 加载 更多

- (void) getmoreData {

if (\_dataArr.count < \_allCount) {

page1++;

[self getDataWithNum:\_page1];

} else {

NSLog(@"没有数据了");

[self endRefreshingFromHeaderOrFooter];

return;

}

}

//选择视图

- (void)initHeaderView {

UIView \*bgView = [[UIView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, 70)];

bgView.backgroundColor = [UIColor whiteColor];

[self.view addSubview:bgView];

UILabel \*startLab = [[UILabel alloc] initWithFrame:CGRectMake(30, 5, 100, 15)];

startLab.text = @"出发地";

startLab.font = [UIFont systemFontOfSize:12];

//UILabel \*startLab1 = [[UILabel alloc] initWithFrame:CGRectMake(30, 20, 100, 30)];

//startLab1.text = @"上海市";

\_startBtn = [self creatbtnWithFrame:CGRectMake(20, 20, 80, 30) withName:@"全国 "];

\_startBtn.tag = 12000;

[\_startBtn addTarget:self action:@selector(choseAddress:) forControlEvents:UIControlEventTouchUpInside];

\_startLab = [[UILabel alloc] initWithFrame:CGRectMake(30, 50, 100, 15)];

\_startLab.text = @"不限";

\_startLab.font = [UIFont systemFontOfSize:12];

UIImageView \*imgView = [[UIImageView alloc]initWithFrame:CGRectMake(kDeviceWidth/2-30, 32, 60, 6)];

imgView.image = [UIImage imageNamed:@"to"];

[bgView addSubview:imgView];

UILabel \*arriveLab = [[UILabel alloc] initWithFrame:CGRectMake(kDeviceWidth - 130, 5, 100, 15)];

arriveLab.text = @"目的地";

arriveLab.textAlignment = NSTextAlignmentRight;

arriveLab.font = [UIFont systemFontOfSize:12];

\_arriveBtn = [self creatbtnWithFrame:CGRectMake(kDeviceWidth - 90, 20, 80, 30) withName:@"全国 "];

\_arriveBtn.tag = 12001;

[\_arriveBtn addTarget:self action:@selector(arriveBtnchoseAddress:) forControlEvents:UIControlEventTouchUpInside];

\_arriveLab = [[UILabel alloc] initWithFrame:CGRectMake(kDeviceWidth - 140, 50, 100, 15)];

\_arriveLab.text = @"不限";

\_arriveLab.font = [UIFont systemFontOfSize:12];

\_arriveLab.textAlignment = NSTextAlignmentRight;

UIView \*lineView = [[UIView alloc]initWithFrame:CGRectMake(0, 69.5, kDeviceWidth, 0.5)];

lineView.backgroundColor = RGBACOLOR(223, 223, 223, 1);

[bgView addSubview:lineView];

[bgView addSubview:startLab];

[bgView addSubview:\_startBtn];

[bgView addSubview:\_startLab];

[bgView addSubview:arriveLab];

[bgView addSubview:\_arriveBtn];

[bgView addSubview:\_arriveLab];

}

- (void)choseAddress:(UIButton \*)choseSender {

\_Type = 1;

[self createCollectionViewWithCity];

}

- (void)arriveBtnchoseAddress:(UIButton \*)arriveBtnchoseAddressSender {

\_Type = 2;

[self createCollectionViewWithCity];

}

#pragma mark -- collection view --

- (void)createCollectionViewWithCity{

\_selectAddressView = [[UIView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight-64)];

\_selectAddressView.backgroundColor = [UIColor whiteColor];

if (\_Type == 1 || \_Type == 2) {

selectAllBtn = [UIButton buttonWithType:UIButtonTypeCustom];

selectAllBtn.frame = CGRectMake(20, 10, 80, 30);

[\_selectAllBtn setTitle:@"全部" forState:UIControlStateNormal];

selectAllBtn.layer.masksToBounds = YES;

selectAllBtn.layer.cornerRadius = 5.0;

selectAllBtn.layer.borderWidth = 1;

selectAllBtn.layer.borderColor = kOrangeColor.CGColor;

[\_selectAllBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

selectAllBtn.backgroundColor = [UIColor whiteColor];

\_selectAllBtn.titleLabel.font = [UIFont systemFontOfSize:14];

[\_selectAllBtn addTarget:self action:@selector(handleSelectAllAction:) forControlEvents:UIControlEventTouchUpInside];

\_returnBtn = [UIButton buttonWithType:UIButtonTypeCustom];

returnBtn.frame = CGRectMake(kDeviceWidth - 100, 10, 80, 30);

[\_returnBtn setTitle:@"返回" forState:UIControlStateNormal];

returnBtn.layer.masksToBounds = YES;

returnBtn.layer.cornerRadius = 5.0;

returnBtn.layer.borderWidth = 1;

returnBtn.layer.borderColor = kOrangeColor.CGColor;

[\_returnBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

returnBtn.backgroundColor = [UIColor whiteColor];

\_returnBtn.titleLabel.font = [UIFont systemFontOfSize:14];

[\_returnBtn addTarget:self action:@selector(handleReturnAllAction:) forControlEvents:UIControlEventTouchUpInside];

\_gobackBtn = [UIButton buttonWithType:UIButtonTypeCustom];

gobackBtn.frame = CGRectMake(120, 10, 80, 30);

[\_gobackBtn setTitle:@"返回上一级" forState:UIControlStateNormal];

gobackBtn.layer.masksToBounds = YES;

gobackBtn.layer.cornerRadius = 5.0;

gobackBtn.layer.borderWidth = 1;

gobackBtn.layer.borderColor = kOrangeColor.CGColor;

[\_gobackBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

gobackBtn.backgroundColor = [UIColor whiteColor];

gobackBtn.titleLabel.font = [UIFont systemFontOfSize:14];

[\_gobackBtn addTarget:self action:@selector(handleGobackAction:) forControlEvents:UIControlEventTouchUpInside];

gobackBtn.hidden = YES;

UIView \*line = [[UIView alloc] initWithFrame:CGRectMake(0, 50, kDeviceWidth, 1)];

line.backgroundColor = [UIColor lightGrayColor];

[\_selectAddressView addSubview:\_selectAllBtn];

[\_selectAddressView addSubview:\_gobackBtn];

[\_selectAddressView addSubview:\_returnBtn];

[\_selectAddressView addSubview:line];

}

CGFloat orignY = 50;

//if (\_Type == 3) {

//UIButton \*filterBtn = [UIButton buttonWithType:UIButtonTypeCustom];

//filterBtn.frame = CGRectMake(20, \_selectAddressView.frame.size.height - 50, kDeviceWidth-40, 30);

//filterBtn.backgroundColor = kOrangeColor;

//filterBtn.layer.masksToBounds = YES;

//filterBtn.layer.cornerRadius = 5.0;

//

//

//[filterBtn setTitle:@"确认选择" forState:UIControlStateNormal];

//[filterBtn setTitleColor:[UIColor whiteColor] forState:UIControlStateNormal];

//filterBtn.titleLabel.font = [UIFont systemFontOfSize:14];

//[filterBtn addTarget:self action:@selector(handleFilterAction:) forControlEvents:UIControlEventTouchUpInside];

//[\_selectAddressView addSubview:filterBtn];

//orignY = 0;

// carsTypeArray = \_lengthArray;

// filterIndex1 = -1;

// filterIndex2 = -1;

//}

UICollectionViewFlowLayout \*layout = [[UICollectionViewFlowLayout alloc] init];

layout.scrollDirection = UICollectionViewScrollDirectionVertical;

//创建UICollectionView对象

self.collectionView = [[UICollectionView alloc] initWithFrame:CGRectMake(0, orignY, kDeviceWidth, kDeviceHeight-64-49) collectionViewLayout:layout];

\_collectionView.backgroundColor = [UIColor clearColor];

\_collectionView.showsVerticalScrollIndicator = NO;

//先要注册Item

[\_collectionView registerClass:[UICollectionViewCell class] forCellWithReuseIdentifier:@"collectionItem"];

[\_collectionView registerClass:[UICollectionReusableView class] forSupplementaryViewOfKind:UICollectionElementKindSectionHeader withReuseIdentifier:@"collectionHeader"];

\_collectionView.delegate = self;

\_collectionView.dataSource = self;

if(\_Type == 3){

//多选的 设置

collectionView.allowsMultipleSelection = YES;

level = 0;

} else {

collectionView.allowsMultipleSelection = NO;

level = 1;

}

[\_selectAddressView addSubview:\_collectionView];

[self.view addSubview:\_selectAddressView];

}

#pragma mark -- collectionView delegate methods --

// item的尺寸

- (CGSize)collectionView:(UICollectionView \*)collectionView layout:(UICollectionViewLayout \*)collectionViewLayout sizeForItemAtIndexPath:(NSIndexPath \*)indexPath

{

return CGSizeMake(90, 35);

}

// header的尺寸

-(CGSize)collectionView:(UICollectionView \*)collectionView layout:(UICollectionViewLayout \*)collectionViewLayout referenceSizeForHeaderInSection:(NSInteger)section{

return CGSizeMake(0, 0);

}

// item缩进

- (UIEdgeInsets)collectionView:(UICollectionView \*)collectionView layout:(UICollectionViewLayout \*)collectionViewLayout insetForSectionAtIndex:(NSInteger)section

{

if (kDeviceWidth == 414) {

return UIEdgeInsetsMake(10, 23, 0, 23);

} else if (kDeviceWidth == 375) {

return UIEdgeInsetsMake(10, 13, 0, 13);

} else {

return UIEdgeInsetsMake(10, 5, 0, 5);

}

}

// 返回分区

- (NSInteger)numberOfSectionsInCollectionView:(UICollectionView \*)collectionView{

return 1;

}

// 分区返回的item

- (NSInteger)collectionView:(UICollectionView \*)collectionView numberOfItemsInSection:(NSInteger)section

{

if (\_level == 1) {

return\_provinces.count;

} else if (\_level == 2) {

return \_cities.count;

}

return 0;

}

- (UICollectionViewCell \*)collectionView:(UICollectionView \*)collectionView cellForItemAtIndexPath:(NSIndexPath \*)indexPath{

static NSString \*itemIndentifier = @"collectionItem";

UICollectionViewCell \*item = [collectionView dequeueReusableCellWithReuseIdentifier:itemIndentifier forIndexPath:indexPath];

if(\_Type == 3) {

item.backgroundColor = [UIColor clearColor];

} else {

item.backgroundColor = [UIColor whiteColor];

item.layer.borderColor = MyCellColor.CGColor;

item.layer.borderWidth = 1.0f;

}

NSString \*name;

if (\_Type == 3) {

} else {

if (\_level == 1) {

name = [\_provinces[indexPath.row] objectForKey:@"state"];

} else if (\_level == 2){

name = [\_cities[indexPath.row] objectForKey:@"city"];

}

}

UILabel \*titleLabel = [[UILabel alloc]initWithFrame:CGRectMake(0, 0, 80, 30)];

titleLabel.font = [UIFont systemFontOfSize:14];

titleLabel.textAlignment = NSTextAlignmentCenter;

[titleLabel sizeToFit];

item.backgroundView = titleLabel;

titleLabel.text = name;

return item;

}

//这个也是最重要的方法 获取Header的 方法。

- (UICollectionReusableView \*)collectionView:(UICollectionView \*)collectionView viewForSupplementaryElementOfKind:(NSString \*)kind atIndexPath:(NSIndexPath \*)indexPath

{

//if (\_Type == 3) {

//NSString \*CellIdentifier = @"collectionHeader";

////从缓存中获取 Headercell

//UICollectionReusableView \*header = [collectionView dequeueReusableSupplementaryViewOfKind:UICollectionElementKindSectionHeader withReuseIdentifier:CellIdentifier forIndexPath:indexPath];

//UILabel \*label = [[UILabel alloc] initWithFrame:CGRectMake(-1, 0, kDeviceWidth+2, 50)];

//label.font = [UIFont systemFontOfSize:14];

//label.layer.borderColor = [UIColor lightGrayColor].CGColor;

//label.layer.borderWidth = 1.0f;

//NSString \*title;

//if (indexPath.section == 0) {

//title = @"车型";

//} else {

//title = @"车长";

//}

//label.text = title;

//[header addSubview:label];

//return header;

//}

return nil;

}

- (void)collectionView:(UICollectionView \*)collectionView didSelectItemAtIndexPath:(NSIndexPath \*)indexPath

{

if (\_Type == 3) {

} else {

// 选取地址

if (\_level == 1) {

// value1 = @"";

// value2 = @"";

choseStr = [[self.provinces objectAtIndex:indexPath.row] objectForKey:@"state"];

cities = [[self.provinces objectAtIndex:indexPath.row] objectForKey:@"cities"];

gobackBtn.hidden = NO;

returnBtn.hidden = YES;

level = 2;

[\_collectionView reloadData];

} else if (\_level == 2) {

value1 = @"";

value2 = @"";

if ([\_choseStr isEqualToString:@"北京"]||[\_choseStr isEqualToString:@"上海"]||[\_choseStr isEqualToString:@"天津"]||[\_choseStr isEqualToString:@"重庆"]) {

value2 = [self.cities[indexPath.row] objectForKey:@"city"];

} else {

value1 = [self.cities[indexPath.row] objectForKey:@"city"];

}

NSLog(@"\_choseStr = %@", \_choseStr);

[self setValueWithDetailAddress:\_Type value1:\_choseStr value2:\_value1 value3:\_value2];

}

}

}

// 选择全部

- (void)handleSelectAllAction:(id)sender{

NSString \*value3 = @"";

if (\_level == 1) {

choseStr = @"全国";

} else if (\_level == 2) {

value3 = \_choseStr;

}

[self setValueWithDetailAddress:\_Type value1:value3 value2:@"" value3:@""];

}

// 返回上一级

- (void)handleGobackAction:(id)sender{

\_level = 1;

\_gobackBtn.hidden = YES;

\_returnBtn.hidden = NO;

[self.collectionView reloadData];

}

//返回

- (void)handleReturnAllAction:(id)sender {

[\_selectAddressView removeFromSuperview];

}

// type 1:始发地 2:目的地

- (void)setValueWithDetailAddress:(NSInteger)type value1:(NSString \*)value1 value2:(NSString \*)value2 value3:(NSString \*)value3{

if (type == 1) {

if ([\_choseStr isEqualToString:@"北京"]||[\_choseStr isEqualToString:@"上海"]||[\_choseStr isEqualToString:@"天津"]||[\_choseStr isEqualToString:@"重庆"]) {

[\_paramDict setValue:value1 forKey:@"cprovince"];

[\_paramDict setValue:value3 forKey:@"ccity"];

} else {

[\_paramDict setValue:value1 forKey:@"cprovince"];

[\_paramDict setValue:value2 forKey:@"ccity"];

}

} else if (type == 2) {

if ([\_choseStr isEqualToString:@"北京"]||[\_choseStr isEqualToString:@"上海"]||[\_choseStr isEqualToString:@"天津"]||[\_choseStr isEqualToString:@"重庆"]) {

[\_paramDict setValue:value1 forKey:@"dprovince"];

[\_paramDict setValue:value3 forKey:@"dcity"];

} else {

[\_paramDict setValue:value1 forKey:@"dprovince"];

[\_paramDict setValue:value2 forKey:@"dcity"];

}

}

\_choseStr = @"";

[\_selectAddressView removeFromSuperview];

NSLog(@"\_paramDict-%@", \_paramDict);

[self setTextAndBtnAndLab];

}

- (void)setTextAndBtnAndLab {

//出发地

NSString \*cp = [NSString stringWithFormat:@"%@", \_paramDict[@"cprovince"]];

NSString \*cc = [NSString stringWithFormat:@"%@", \_paramDict[@"ccity"]];

NSLog(@"cp-%@---cc = %@---", cp, cc);

if ([cp isEqualToString:@""]) {

[\_startBtn setTitle:@"全国" forState:UIControlStateNormal];

} else {

[\_startBtn setTitle:cp forState:UIControlStateNormal];

}

if ([cc isEqualToString:@""]) {

[\_startLab setText:@"不限"];

} else {

[\_startLab setText:cc];

}

//目的地

NSString \*dp = [NSString stringWithFormat:@"%@", \_paramDict[@"dprovince"]];

NSString \*dc = [NSString stringWithFormat:@"%@", \_paramDict[@"dcity"]];

NSLog(@"dp-%@---dc = %@---", dp, dc);

if ([dp isEqualToString:@""]) {

[\_arriveBtn setTitle:@"全国" forState:UIControlStateNormal];

} else {

[\_arriveBtn setTitle:dp forState:UIControlStateNormal];

}

if ([dc isEqualToString:@""]) {

[\_arriveLab setText:@"不限"];

} else {

[\_arriveLab setText:dc];

}

[\_dataArr removeAllObjects];

\_page1=0;

[self getDataWithNum:\_page1];

}

#pragma mark----数据优化

- (void)getDataWithNum:(int ) num {

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dict = [[NSUserDefaults standardUserDefaults] objectForKey:@"user"];

NSString \*lat = [NSString stringWithFormat:@"%@", \_paramDict[@"x"]];

NSString \*lng = [NSString stringWithFormat:@"%@", \_paramDict[@"y"]];

NSString \*type = @"附近车源";

NSString \*ctype = @"";

NSString \*userName = [NSString stringWithFormat:@"%@", dict[@"username"]];

NSString \*rank = [NSString stringWithFormat:@"%@", \_paramDict[@"rank"]];

NSString \*cprovince = [NSString stringWithFormat:@"%@", \_paramDict[@"cprovince"]];

NSString \*ccity = [NSString stringWithFormat:@"%@", \_paramDict[@"ccity"]];

NSString \*dprovince = [NSString stringWithFormat:@"%@", \_paramDict[@"dprovince"]];

NSString \*dcity = [NSString stringWithFormat:@"%@", \_paramDict[@"dcity"]];

NSString \*page = [NSString stringWithFormat:@"%d", num];

NSLog(@"lat = %@--lng = %@--type = %@---ctype = %@---userName = %@---cprovince = %@--ccity = %@-----dprovince = %@----dcity = %@ -rank=%@", lat, lng, type, ctype, userName, cprovince, ccity, dprovince, dcity, rank);

NSDictionary \*parma = [NSDictionary dictionaryWithObjectsAndKeys:@"", @"ctype" ,userName, @"username", lat, @"x", lng, @"y", type, @"type", rank, @"rank", cprovince, @"cprovince", ccity, @"ccity", dprovince, @"dprovince", dcity, @"dcity", page, @"page",nil];

NSLog(@"新的字典= %@", parma);

[BJHttpTool post:Knewapi3GetcarfbListByUser3 parameters:parma success:^(id responseObject) {

if (responseObject) {

[self endRefresh];

NSError \*error = nil;

NSMutableDictionary \*rootDict = [[NSMutableDictionary alloc] init];

rootDict = [NSJSONSerialization JSONObjectWithData:responseObject options:0 error:&error];

NSLog(@"rootDict = %@",rootDict);

if ([[rootDict objectForKey:@"errcode"] intValue] == 0) {

[lbxManager showHudViewLabelText:@"获取附近业务成功" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

allCount = [[NSString stringWithFormat:@"%@", [NSString stringWithFormat:@"%@", rootDict[@"count"]]] integerValue];

for (NSDictionary \*dic in rootDict[@"list"]) {

HYDDoodsModel \*model = [[HYDDoodsModel alloc]init];

model.username = [dic objectForKey:@"czuser"];

model.daddress = [NSString stringWithFormat:@"%@%@%@%@",[dic objectForKey:@"dprovince"],[dic objectForKey:@"dcity"],[dic objectForKey:@"darea"],[dic objectForKey:@"daddress"]];

model.je = [dic objectForKey:@"price"];

model.dstrtime = @"";

model.paytype = @"";

model.cjfs = @"";

model.caddress = [NSString stringWithFormat:@"%@%@%@%@",[dic objectForKey:@"cprovince"],[dic objectForKey:@"ccity"],[dic objectForKey:@"carea"],[dic objectForKey:@"caddress"]];

model.mytb = @"";

model.ystype = [dic objectForKey:@"csize"];

model.zl = [dic objectForKey:@"cartype"];

model.cstrtime = [dic objectForKey:@"strtime"];

model.name = [dic objectForKey:@"name"];

model.orderid = @"";

model.Atype = @"货主筛选";

model.latitude = [dic objectForKey:@"latitude"];

model.stime = [dic objectForKey:@"starttime"];

model.longitude = [dic objectForKey:@"longitude"];

model.product = [dic objectForKey:@"cph"];

model.ttime = [dic objectForKey:@"starttime"];

model.headimg = [dic objectForKey:@"headimg"];

model.sl = [dic objectForKey:@"carfbid"];

[\_dataArr addObject:model];

}

if (self.dataArr.count>=\_allCount) {

[lbxManager showHudViewLabelText:@"附近车源已全部加载" detailsLabelText:nil afterDelay:kAfterDelayTime];

self.myOrderList.footer.state = MJRefreshStateNoMoreData;

}

} else {

NSLog(@"数据请求错误!");

}

}

[\_sorryLabel setHidden:\_dataArr.count];

[\_myOrderList reloadData];

} failure:^(NSError \*error) {

[self endRefreshingFromHeaderOrFooter];

}];

}

//底部视图

- (void) initBootomView {

UIView \*aView = [[UIView alloc] init];

aView.frame = CGRectMake(0, kDeviceHeight - 64 - 40, kDeviceWidth, 40);

aView.backgroundColor = [UIColor whiteColor];

[self.view addSubview:aView];

UIView \*lineView = [[UIView alloc] init];

lineView.frame = CGRectMake(0, 1, kDeviceWidth, 1);

lineView.backgroundColor = MyLineColor;

[aView addSubview:lineView];

//\_choseTypeBtn = [UIButton buttonWithType:UIButtonTypeCustom];

//\_choseTypeBtn.frame = CGRectMake(15, 5, 150, 30);

//\_choseTypeBtn.contentHorizontalAlignment = UIControlContentHorizontalAlignmentLeft;

//\_choseTypeBtn.backgroundColor = [UIColor orangeColor];

//[\_choseTypeBtn setTitle:@"离我最近" forState:UIControlStateNormal];

//[\_choseTypeBtn addTarget:self action:@selector(choseTypeAction:) forControlEvents:UIControlEventTouchUpInside];

//[aView addSubview:\_choseTypeBtn];

\_choseTypeBtn = [self creatbtnWithFrame:CGRectMake(15, 5, 100, 30) withName:@"离我最近 "];

//\_choseTypeBtn.layer.borderWidth = 1.0;

//\_choseTypeBtn.layer.borderColor = kOrangeColor.CGColor;

[\_choseTypeBtn addTarget:self action:@selector(choseTypeAction:) forControlEvents:UIControlEventTouchUpInside];

[aView addSubview:\_choseTypeBtn];

UIButton \*mapBtn = [UIButton buttonWithType:UIButtonTypeCustom];

mapBtn.frame = CGRectMake(kDeviceWidth - 100, 5, 90, 30);

mapBtn.backgroundColor = [UIColor whiteColor];

mapBtn.layer.borderColor = kOrangeColor.CGColor;

mapBtn.layer.borderWidth = 1.0f;

mapBtn.layer.masksToBounds = YES;

mapBtn.layer.cornerRadius = 5.0;

mapBtn.titleLabel.font = [UIFont systemFontOfSize:15];

[mapBtn setTitle:@"终点地图" forState:UIControlStateNormal];

[mapBtn setTitleColor:kOrangeColor forState:UIControlStateNormal];

[mapBtn setImage:[UIImage imageNamed:@"map-"] forState:UIControlStateNormal];

//图片左边

CGSize imageSize = mapBtn.imageView.frame.size;

mapBtn.titleEdgeInsets = UIEdgeInsetsMake(0.0, 0.0, 0.0, -imageSize.width);

[mapBtn addTarget:self action:@selector(seeMapAction:) forControlEvents:UIControlEventTouchUpInside];

[aView addSubview:mapBtn];

}

//筛选视图

- (void)creatChoseView {

\_isChoseExist = true;

\_smallView = [[UIView alloc] init];

\_smallView.backgroundColor = RGBACOLOR(230, 230, 230, 1);

\_smallView.frame = CGRectMake(0, kDeviceHeight - 64 - 40 - 105, kDeviceWidth, 105);

[self.view addSubview:\_smallView];

//距离最近

UIButton \*btn1 = [UIButton buttonWithType:UIButtonTypeCustom];

btn1.frame = CGRectMake(15, 0, 90, 35);

[btn1 setTitle:@"离我最近" forState:UIControlStateNormal];

btn1.contentHorizontalAlignment = UIControlContentHorizontalAlignmentCenter;

[btn1 setTitleColor:[UIColor blackColor] forState:UIControlStateNormal];

[btn1 addTarget:self action:@selector(btn1Action:) forControlEvents:UIControlEventTouchUpInside];

UIView \*lineView1 = [[UIView alloc] init];

lineView1.frame = CGRectMake(15, 35, 90, 1);

lineView1.backgroundColor = [UIColor lightGrayColor];

[\_smallView addSubview:lineView1];

[\_smallView addSubview:btn1];

//出发最快

UIButton \*btn2 = [UIButton buttonWithType:UIButtonTypeCustom];

btn2.frame = CGRectMake(15, 36, 90, 35);

[btn2 setTitle:@"最快出发" forState:UIControlStateNormal];

btn2.contentHorizontalAlignment = UIControlContentHorizontalAlignmentCenter;

[btn2 setTitleColor:[UIColor blackColor] forState:UIControlStateNormal];

[btn2 addTarget:self action:@selector(btn2Action:) forControlEvents:UIControlEventTouchUpInside];

UIView \*lineView2 = [[UIView alloc] init];

lineView2.frame = CGRectMake(15, 71, 90, 1);

lineView2.backgroundColor = [UIColor lightGrayColor];

[\_smallView addSubview:lineView2];

[\_smallView addSubview:btn2];

//费用最高

UIButton \*btn3 = [UIButton buttonWithType:UIButtonTypeCustom];

btn3.frame = CGRectMake(15, 71, 90, 35);

[btn3 setTitle:@"费用最高" forState:UIControlStateNormal];

btn3.contentHorizontalAlignment = UIControlContentHorizontalAlignmentCenter;

[btn3 setTitleColor:[UIColor blackColor] forState:UIControlStateNormal];

[btn3 addTarget:self action:@selector(btn3Action:) forControlEvents:UIControlEventTouchUpInside];

[\_smallView addSubview:btn3];

}

- (void)mapList:(id)sender1 {

if (\_dataArr.count != 0) {

BJMyNearbyBusinessViewController \*myBussOnMapVC = [[BJMyNearbyBusinessViewController alloc] init];

myBussOnMapVC.userType = \_userType;

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

} else {

ALERTSHOWREMIND(@"暂无附件业务");

}

}

#pragma mark---- UITableViewDataSource

- (CGFloat)tableView:(UITableView \*)tableView heightForRowAtIndexPath:(NSIndexPath \*)indexPath {

return 170;

}

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section {

return \_dataArr.count;

}

- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath {

static NSString \*identifer11 = @"myListCell";

HYDGoogListCell \*myListCell = [tableView dequeueReusableCellWithIdentifier:identifer11];

if (!myListCell) {

myListCell = [[HYDGoogListCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:identifer11];

}

myListCell.selectionStyle = UITableViewCellSelectionStyleNone;

HYDDoodsModel \*model = \_dataArr[indexPath.row];

myListCell.model = model;

myListCell.dImgView.hidden = YES;

[myListCell.btn setTitle:@"约车" forState:UIControlStateNormal];

[myListCell.btn addTarget:self action:@selector(evaluteAction:event:) forControlEvents:UIControlEventTouchUpInside];

return myListCell;

}

- (void)tableView:(UITableView \*)tableView didSelectRowAtIndexPath:(NSIndexPath \*)indexPath {

if (\_isChoseExist == true) {

isChoseExist = false;

[\_smallView removeFromSuperview];

}

MyCustomCarsInfoWithMapVC \*myCarInfoVC = [[MyCustomCarsInfoWithMapVC alloc] init];

HYDDoodsModel \*model = \_dataArr[indexPath.row];

myCarInfoVC.carID = model.sl;

myCarInfoVC.phoneStr = model.username;

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

}

#pragma mark------Action

//筛选

- (void)searchClick {

NSLog(@"筛选");

//车辆搜索

HomeChildController \*searchGoodsVC = [[HomeChildController alloc] init];

searchGoodsVC.userType = @"货主";

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

}

- (void)choseTypeAction:(UIButton \*)sender1 {

if (\_isChoseExist == false) {

[self creatChoseView];

} else {

NSLog(@"已经存在！！");

isChoseExist = false;

[\_smallView removeFromSuperview];

}

}

- (void)seeMapAction:(UIButton \*)sender2 {

NSLog(@"终点地图！！！");

if (\_dataArr.count == 0) {

ALERTSHOWREMIND(@"暂无附近业务");

return;

} else {

BJMyNearbyBusinessViewController \*nearbyMapVC = [[BJMyNearbyBusinessViewController alloc] init];

nearbyMapVC.userType = @"客户";

nearbyMapVC.cprovice = @"";

nearbyMapVC.ccity = @"";

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

}

}

- (void)btn1Action:(UIButton \*) sender3 {

[\_choseTypeBtn setTitle:@"离我最近" forState:UIControlStateNormal];

[self choseTypeStr:@"离我最近"];

[\_smallView removeFromSuperview];

}

- (void)btn2Action:(UIButton \*) sender4 {

[\_choseTypeBtn setTitle:@"最快出发" forState:UIControlStateNormal];

[self choseTypeStr:@"最快出发"];

[\_smallView removeFromSuperview];

}

- (void)btn3Action:(UIButton \*) sender5 {

[\_choseTypeBtn setTitle:@"费用最高" forState:UIControlStateNormal];

[self choseTypeStr:@"费用最高"];

[\_smallView removeFromSuperview];

}

//选择方式

- (void)choseTypeStr:(NSString \*) typeStr {

\_isChoseExist = false;

NSLog(@"什么鬼= %@", typeStr);

[\_paramDict setValue:typeStr forKey:@"rank"];

if ([typeStr isEqualToString:@"离我最近"]) {

NSString \*lat = [NSString stringWithFormat:@"%f", \_lat];

NSString \*lng = [NSString stringWithFormat:@"%f", \_lng];

[\_paramDict setValue:lat forKey:@"x"]; //经度

[\_paramDict setValue:lng forKey:@"y"]; //纬度

} else {

[\_paramDict setValue:@"" forKey:@"x"]; //经度

[\_paramDict setValue:@"" forKey:@"y"]; //纬度

}

[\_dataArr removeAllObjects];

[self getDataWithNum:\_page1];

}

- (void)evaluteAction:(id)sender1 event:(id)event {

NSLog(@"跳转！！！！");

NSSet \*touches =[event allTouches];

UITouch \*touch =[touches anyObject];

CGPoint currentTouchPosition = [touch locationInView:\_myOrderList];

NSIndexPath \* indexPath = [\_myOrderList indexPathForRowAtPoint:currentTouchPosition];

MyCustomCarsInfoWithMapVC \*myCarInfoVC = [[MyCustomCarsInfoWithMapVC alloc] init];

HYDDoodsModel \*model = \_dataArr[indexPath.row];

myCarInfoVC.carID = model.sl;

myCarInfoVC.phoneStr = model.username;

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

}

/\*\*

\*停止刷新

\*/

-(void)endRefresh{

[self.myOrderList.header endRefreshing];

}

- (UIButton \*)creatbtnWithFrame:(CGRect)frame withName:(NSString \*)name {

//UIButton \*btn = [UIButton buttonWithType:UIButtonTypeCustom];

LZRelayoutButton \*btn = [LZRelayoutButton buttonWithType:UIButtonTypeCustom];

btn.frame = frame;

[btn setTitle:name forState:UIControlStateNormal];

btn.lzType = LZRelayoutButtonTypeLeft;

[btn setTitleColor:[UIColor blackColor] forState:UIControlStateNormal];

[btn setTitleColor:[UIColor redColor] forState:UIControlStateSelected];

[btn setImage:[UIImage imageNamed:@"down\_grey"] forState:UIControlStateNormal];

[btn setImage:[UIImage imageNamed:@"down\_grey"] forState:UIControlStateSelected];

btn.titleLabel.font = [UIFont systemFontOfSize:15];

return btn;

}

- (void)setExtraCellLineHidden:(UITableView \*)tableView {

UIView \*view = [UIView new];

view.backgroundColor = [UIColor clearColor];

[tableView setTableFooterView:view];

}

-(void)viewDidLayoutSubviews {

if ([\_myOrderList respondsToSelector:@selector(setSeparatorInset:)]) {

[\_myOrderList setSeparatorInset:UIEdgeInsetsZero];

}

if ([\_myOrderList respondsToSelector:@selector(setLayoutMargins:)]){

[\_myOrderList setLayoutMargins:UIEdgeInsetsZero];

}

}

-(void)tableView:(UITableView \*)tableView willDisplayCell:(UITableViewCell \*)cell forRowAtIndexPath:(NSIndexPath \*)indexPat{

if ([cell respondsToSelector:@selector(setLayoutMargins:)]) {

[cell setLayoutMargins:UIEdgeInsetsZero];

}

if ([cell respondsToSelector:@selector(setSeparatorInset:)]){

[cell setSeparatorInset:UIEdgeInsetsZero];

}

}

// 结束刷新

- (void)endRefreshingFromHeaderOrFooter

{

NSLog(@"结束刷新了啊刷新了啊");

[\_myOrderList.footer endRefreshing];

[\_myOrderList.header endRefreshing];

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}

#import "HYDNearCarMapVC.h"

#import "HomeChildController.h"

#import "HYDNearCarModel.h"

#import "BJCarAnnotation.h"

#import "HYDAnnotationView.h"

#import "HYDGPSpaopaoView.h"

#import "PublicHelp.h"

#import "PublishProducInfotVC.h"

#import "HYDNearCarDetailView.h"

@interface HYDNearCarMapVC ()<BMKMapViewDelegate>

{

BMKMapView \*\_mapView;

}

@property (nonatomic , strong)NSMutableArray \*annotionArr;

@property (nonatomic , strong)NSMutableArray \*carListArr;

@property (nonatomic , strong) HYDNearCarDetailView \*detailView;

@end

@implementation HYDNearCarMapVC

- (void)viewDidLoad {

[super viewDidLoad];

self.title = @"搜索车源";

self.view.backgroundColor = [UIColor whiteColor];

[self creatRightBtnItem];

[self setupMapViewWithParam];

[self initData];

}

#pragma mark - 设置百度地图

-(void)setupMapViewWithParam {

//初始化BMKMapView

\_mapView = [[BMKMapView alloc] initWithFrame:CGRectMake(0, 0, kDeviceWidth, kDeviceHeight - 64)];

\_mapView.delegate = self;

\_mapView.zoomLevel = 5;

\_mapView.showsUserLocation = YES;

[\_mapView addSubview:self.detailView];

[self.view addSubview:\_mapView];

}

- (void)dealloc{

\_mapView.delegate = nil;

\_mapView = nil;

}

- (void)initData

{

LBXManager \*lbxManager = [LBXManager sharedManager];

if ([lbxManager checkIsHasNetwork] == NO) {

return;

}

[lbxManager showHUDAction:0 showView:self.view];

NSDictionary \*dic = [NSDictionary dictionaryWithObjectsAndKeys:[NSString stringWithFormat:@"%f",\_lat],@"lat",[NSString stringWithFormat:@"%f",\_lng],@"lon",@"JHSW",@"username",@"150",@"range", nil];

[[BJHttpClient shareInstance]postRequestWithUrl:KGetNearCar withParam:dic withView:self.view withText:@"数据请求中" sucessBlock:^(id data) {

NSLog(@"%@",data);

if ([[data objectForKey:@"errcode"] intValue]==0) {

NSArray \*carArr = [data objectForKey:@"carList"];

if (carArr.count==0) {

[lbxManager showHudViewLabelText:@"附近暂无车辆" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}else{

for (NSDictionary \*dict in carArr) {

HYDNearCarModel \*model = [HYDNearCarModel objecWithKeyValues:dict];

[self.carListArr addObject:model];

}

[self creatAnnotionView];

}

}else{

[lbxManager showHudViewLabelText:[data objectForKey:@"msg"] detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}

} faileBlock:^(NSString \*error) {

[lbxManager showHudViewLabelText:@"获取信息失败 请稍后再试" detailsLabelText:nil afterDelay:kAfterDelayTime];

[lbxManager showHUDAction:1 showView:self.view];

}];

[lbxManager showHUDAction:1 showView:self.view];

}

- (void)creatRightBtnItem

{

//筛选按钮

UIBarButtonItem \*searchItem = [[UIBarButtonItem alloc] initWithTitle:@"筛选" style:UIBarButtonItemStylePlain target:self action:@selector(searchClick)];

searchItem.tintColor = [UIColor whiteColor];

self.navigationItem.rightBarButtonItem = searchItem;

}

- (void)searchClick

{

NSLog(@"筛选");

//货源搜索

HomeChildController \*searchGoodsVC = [[HomeChildController alloc] init];

searchGoodsVC.userType = @"货主";

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

}

- (void)creatAnnotionView

{

CLLocationCoordinate2D paths[self.carListArr.count];

for (NSInteger i=0; i<self.carListArr.count; i++) {

HYDNearCarModel \*model = self.carListArr[i];

BJCarAnnotation \*carAnnotationModel = [[BJCarAnnotation alloc]init];

carAnnotationModel.cph = model.cph;

carAnnotationModel.ISG = model.ISG;

carAnnotationModel.Gcph = model.Gcph;

carAnnotationModel.Gbrand = model.Gbrand;

carAnnotationModel.Gzs = model.Gzs;

carAnnotationModel.Gzh = model.Gzh;

carAnnotationModel.driver = model.driver;

carAnnotationModel.cartype = model.cartype;

carAnnotationModel.isdanger = model.isdanger;

carAnnotationModel.tel = model.tel;

carAnnotationModel.tboxid = model.tboxid;

carAnnotationModel.speed = @"0";

CLLocationCoordinate2D carCoordinate = CLLocationCoordinate2DMake([model.lat floatValue], [model.lon floatValue]);

carAnnotationModel.coordinate = carCoordinate;

[self.annotionArr addObject:carAnnotationModel];

paths[i] = carCoordinate;

}

[\_mapView addAnnotations:self.annotionArr];

BMKPolyline \*linepolyLine = [BMKPolyline polylineWithCoordinates:paths count:self.carListArr.count];

[self mapViewFitPolyLine:linepolyLine];

}

- (void)mapViewFitPolyLine:(BMKPolyline \*) polyLine {

CGFloat ltX, ltY, rbX, rbY;

if (polyLine.pointCount < 1) {

return;

}

BMKMapPoint pt = polyLine.points[0];

ltX = pt.x, ltY = pt.y;

rbX = pt.x, rbY = pt.y;

for (int i = 1; i < polyLine.pointCount; i++) {

BMKMapPoint pt = polyLine.points[i];

if (pt.x < ltX) {

ltX = pt.x;

}

if (pt.x > rbX) {

rbX = pt.x;

}

if (pt.y > ltY) {

ltY = pt.y;

}

if (pt.y < rbY) {

rbY = pt.y;

}

}

BMKMapRect rect;

rect.origin = BMKMapPointMake(ltX , ltY);

rect.size = BMKMapSizeMake(rbX - ltX, rbY - ltY);

[\_mapView setVisibleMapRect:rect];

\_mapView.zoomLevel = \_mapView.zoomLevel - 0.3;

}

#pragma mark -BMKMapViewDelegate

- (MKAnnotationView \*)mapView:(MKMapView \*)mapView viewForAnnotation:(id<MKAnnotation>)annotation {

// 创建大头针

if ([annotation isKindOfClass:[BJCarAnnotation class]]) {

BMKPinAnnotationView \*newAnnotationView = [[BMKPinAnnotationView alloc] initWithAnnotation:annotation reuseIdentifier:@"myAnnotation"];

newAnnotationView.animatesDrop = YES;// 设置该标注点动画显示

BJCarAnnotation \*carAnnotion = annotation;

if (carAnnotion.Gzs.length==0) {

newAnnotationView.image = [UIImage imageNamed:@"carstand"];

}else{

HYDAnnotationView \*anView = [[HYDAnnotationView alloc]initWithCarAnnotation:carAnnotion];

newAnnotationView.image = [anView getImageFromView];

}

HYDGPSpaopaoView \*paopao = [[HYDGPSpaopaoView alloc]initWithCarAnnotation:carAnnotion];

newAnnotationView.paopaoView = [[BMKActionPaopaoView alloc]initWithCustomView:paopao];

return newAnnotationView;

}

return nil;

}

#pragma mark----- 点击大头针

- (void)mapView:(BMKMapView \*)mapView didSelectAnnotationView:(BMKAnnotationView \*)view{

NSLog(@"经度 =%f-----纬度=%f", view.annotation.coordinate.latitude, view.annotation.coordinate.longitude);

BJCarAnnotation \*carAnnotion = (BJCarAnnotation \*)view.annotation;

CLLocationCoordinate2D pt = (CLLocationCoordinate2D){0, 0};

pt = (CLLocationCoordinate2D){view.annotation.coordinate.latitude, view.annotation.coordinate.longitude};

NSLog(@"%@", carAnnotion.cph);

[\_mapView setCenterCoordinate:pt];

NSMutableArray \*arr = [[NSMutableArray alloc]init];

for (NSInteger i=0; i<self.carListArr.count; i++) {

HYDNearCarModel \*model = self.carListArr[i];

[arr addObject:model.tel];

}

for (NSInteger i=0; i<self.carListArr.count; i++) {

HYDNearCarModel \*model = self.carListArr[i];

if ([model.cph isEqualToString:carAnnotion.cph]) {

[self.detailView showWithModel:model];

PublishProducInfotVC \*publishVC = [[PublishProducInfotVC alloc] init];

//此处传空。方便草稿重发 和 失效重发的处理!

NSUserDefaults \*user = [NSUserDefaults standardUserDefaults];

NSString \*orderid = [user valueForKey:@"orderid"];

[user setObject:[arr componentsJoinedByString:@","] forKey:@"yueche"];

[user synchronize];

publishVC.orderID = [[orderid componentsSeparatedByString:@"-"] firstObject];

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

return;

}

}

}

- (void)mapView:(BMKMapView \*)mapView onClickedMapBlank:(CLLocationCoordinate2D)coordinate

{

[self.detailView close];

}

- (NSMutableArray \*)annotionArr

{

if (!\_annotionArr) {

annotionArr = [[NSMutableArray alloc]init];

}

return \_annotionArr;

}

- (NSMutableArray \*)carListArr

{

if (!\_carListArr) {

carListArr = [[NSMutableArray alloc]init];

}

return \_carListArr;

}

- (HYDNearCarDetailView \*)detailView

{

if (!\_detailView) {

detailView = [[HYDNearCarDetailView alloc]initWithFrame:CGRectMake(0, kDeviceHeight, kDeviceWidth, 110)];

}

return \_detailView;

}