#import "HomeVC.h"

#import "HomeTopView.h"

#import "HomeBottomView.h"

#import "NSString+size.h"

#import "HomeSchoolModel.h"

@interface HomeVC ()

@property (nonatomic,strong) UIScrollView \*scrollView;

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

@end

@implementation HomeVC

- (void)viewDidLoad {

[super viewDidLoad];

self.view.backgroundColor = OVERALLCOLOR;

self.navigationItem.leftBarButtonItem = [UIBarButtonItem barButtonItemWithTitle:@"首页" titleColor:[UIColor whiteColor] andBarButtonItemBlock:^{

}];

self.scrollView = [[UIScrollView alloc] initWithFrame:self.view.bounds];

WEAKSELF

[self.scrollView addHeaderRefresh:^{

[weakSelf\_SC reqeustData];

[weakSelf\_SC.scrollView endHeaderRefresh];

}];

// self.scrollView.bounces = NO;

[self.view addSubview:self.scrollView];

HomeTopView \*topView = [HomeTopView viewFromNIB];

//点击组卷

topView.clickOrganizeBlock = ^{

[ROOTTABBARVC setSelectedIndex:1];

};

//点击收藏

topView.clickCollectionBlock = ^{

kPushStoryboardVC(@"Home", @"HomeCollectTVCSBID");

};

[self.scrollView addSubview:topView];

HomeBottomView \*bottomView = [HomeBottomView viewFromNIB];

bottomView.frame = CGRectMake(0, CGRectGetMaxY(topView.frame), SCREEN\_WIDTH, SCREEN\_HEIGHT - topView.height - 49);

self.bottomView = bottomView;

[self.scrollView addSubview:bottomView];

[self reqeustData];

}

- (void)didReceiveMemoryWarning {

[super didReceiveMemoryWarning];

// Dispose of any resources that can be recreated.

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

- (void)reqeustData

{

[JXHttpTool POST:@"schoolShow" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber?:EMPTY\_STRING} success:^(id responseObject) {

HomeSchoolModel \*model = [HomeSchoolModel parse:responseObject[@"SchoolInfo"]];

self.bottomView.schoolName.text = model.SchoolName;

NSString \*schoolIntro = model.Introduction;

[self.bottomView.icon sd\_setImageWithURL:URLWITHSTRING(responseObject[model.LogoUrl?:EMPTY\_STRING]) placeholderImage:IMG(@"")];

[self.bottomView refreshWithIntro:schoolIntro andTitleHieght:self.bottomView.height - 169 - 35 configScrollView:self.scrollView];

} failure:nil];

}

@end

#import "HomeCollectTVC.h"

#import "HomeCollectCreateAndEdit.h"

#import "HomeCollectCell.h"

#import "PaperBrowseVC.h"

@interface HomeCollectTVC ()

@end

@implementation HomeCollectTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"icon\_add" andBarButtonItemBlock:^{

HomeCollectCreateAndEdit \*collectionVC = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

collectionVC.isAddCollect = YES;

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

}];

WEAKSELF

kTableViewFuckingInit(weakSelf\_SC.tableView,requestCollectData);

}

- (void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

[self requestCollectData];

}

- (void)requestCollectData

{

[JXHttpTool POST:@"getFavoritesList" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"10"} success:^(id responseObject) {

kTableViewAfterDataAction(self.tableView,responseObject,nil);

} failure:^(id error) {

NSLog(@"%@",error);

} tableView:self.tableView];

}

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

{

return self.tableView.tableViewDatas.count;

}

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

{

HomeCollectCell \*cell = [HomeCollectCell cellWithNibTableView:tableView andReuseID:@"HomeCollectCellID"];

cell.clickEditBlock = ^{

HomeCollectCreateAndEdit \*collectionVC = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

collectionVC.isEditCollect = YES;

collectionVC.inPlatformId = self.tableView.tableViewDatas[indexPath.row][@"PlatformId"];

collectionVC.collectID = self.tableView.tableViewDatas[indexPath.row][@"Id"];

collectionVC.collectBeginName = self.tableView.tableViewDatas[indexPath.row][@"FavoritesName"];

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

};

cell.name.text = self.tableView.tableViewDatas[indexPath.row][@"FavoritesName"];

cell.num.text = [NSString stringWithFormat:@"%@",self.tableView.tableViewDatas[indexPath.row][@"SumCount"]];

return cell;

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if ([[NSString stringWithFormat:@"%@",self.tableView.tableViewDatas[indexPath.row][@"SumCount"]] isEqualToString:@"0"]) {

SHOWERROR(@"暂无试题");

return;

}

PaperBrowseVC \*paperBrowseVC = [[PaperBrowseVC alloc] init];

paperBrowseVC.inPlatformId = self.tableView.tableViewDatas[indexPath.row][@"PlatformId"];

paperBrowseVC.inFavId = self.tableView.tableViewDatas[indexPath.row][@"Id"];

paperBrowseVC.allNum = [NSString stringWithFormat:@"%@",self.tableView.tableViewDatas[indexPath.row][@"SumCount"]];

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

}

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

{

return 80;

}

@end

//

// HomeCollectCreateAndEdit.m

// MaestroEdu

//

// Created by 张明辉 on 2017/7/10.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "HomeCollectCreateAndEdit.h"

#import "HomeAddRecordBarView.h"

#import "BDImagePicker.h"

@interface HomeCollectCreateAndEdit ()

{

NSMutableArray \*\_iconArr;

NSMutableArray \*\_imageViewArr;

NSMutableArray \*\_iconDataArr;

}

@property (weak, nonatomic) IBOutlet UITextView \*textView;

@property (weak, nonatomic) IBOutlet UITextField \*topTextField;

@property (weak, nonatomic) IBOutlet NSLayoutConstraint \*topTextFieldHeight;

@property (weak, nonatomic) IBOutlet UIButton \*deleteBtn;

@property (weak, nonatomic) IBOutlet UIView \*imagesBgView;

@property (weak, nonatomic) IBOutlet UIImageView \*icon1;

@property (weak, nonatomic) IBOutlet UIImageView \*icon2;

@property (weak, nonatomic) IBOutlet UIImageView \*icon3;

@property (weak, nonatomic) IBOutlet UIImageView \*icon4;

@property (weak, nonatomic) IBOutlet UIImageView \*icon5;

@property (weak, nonatomic) IBOutlet UIImageView \*icon6;

@end

@implementation HomeCollectCreateAndEdit

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

\_iconArr = @[].mutableCopy;

\_imageViewArr = @[self.icon1,self.icon2,self.icon3,self.icon4,self.icon5,self.icon6].mutableCopy;

\_iconDataArr = @[].mutableCopy;

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"general\_icon\_right" andBarButtonItemBlock:^{

[self updateData];

}];

self.topTextFieldHeight.constant = 0;

self.deleteBtn.hidden = YES;

self.textView.placeholder = @"请输入10字以内的收藏夹名称";

if (self.isAddCollect) {

self.title = @"创建收藏夹";

self.deleteBtn.hidden = YES;

} else if (self.isEditCollect) {

self.title = @"编辑收藏夹";

self.deleteBtn.hidden = NO;

self.textView.text = self.collectBeginName;

}

if (self.isClassessIntro) {

//是班级简介的话

self.title = @"班级简介";

self.textView.text = self.classessIntroStr;

self.textView.placeholder = @"请输入200字以内的班级简介";

} else if (self.isClassessPublicNotice) {

//是班级公告的话

self.title = @"班级公告";

self.topTextFieldHeight.constant = 30;

self.textView.placeholder = @"请输入15-500字的班级公告";

self.topTextField.text = self.publicNoticeTitle;

self.textView.text = self.publicNoticeDetail;

}

if (self.isAddRecord) {

//添加笔记的话

self.title = @"添加笔记";

self.textView.text = EMPTY\_STRING;

self.textView.placeholder = @"请输入300字以内的笔记";

// [IQKeyboardManager sharedManager]

HomeAddRecordBarView \*barView = [HomeAddRecordBarView viewFromNIB];

barView.clickPhoteBlock = ^{

//调用相机

[BDImagePicker showImagePickerFromViewController:self allowsEditing:YES finishAction:^(UIImage \*image) {

if (image) {

[self refreshImageWith:image];

}

} isAddRecord:1];

};

barView.clickPictureBlock = ^{

//调用相册

[BDImagePicker showImagePickerFromViewController:self allowsEditing:YES finishAction:^(UIImage \*image) {

if (image) {

[self refreshImageWith:image];

}

} isAddRecord:2];

};

self.textView.inputAccessoryView = barView;

}

if (self.isModifyInfo) {

//修改个人信息的话

self.title = @"修改介绍";

self.textView.text = self.para[@"inIntroduce"];

self.textView.placeholder = @"请输入50字以内的介绍";

}

}

- (void)refreshImageWith:(UIImage \*)image

{

if (\_iconArr.count == 6) {

SHOWERROR(@"最多只能上传六张");

return;

}

NSData \*imageData = kImageDataCompress(image);

[\_iconDataArr addObject:imageData];

[\_iconArr addObject:image];

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

[\_imageViewArr[i] setImage:\_iconArr[i]];

}

}

- (IBAction)clickDeleteBtn:(UIButton \*)sender {

[UIAlertTool showAlertViewWithTitle:@"提示" message:@"确定删除该收藏夹吗?" cancelButtonTitle:@"取消" otherButtonTitle:@"确定" confirm:^{

[JXHttpTool POST:@"deleteFavorites" parameters:@{@"inPlatformId":self.inPlatformId?:EMPTY\_STRING,@"inId":self.collectID?:EMPTY\_STRING} success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:nil];

} cancle:nil];

}

- (void)updateData {

if (self.isClassessIntro) {

//班级简介

kJudgeNullWith(self.textView.text, @"请输入班级简介");

if (self.textView.text.length > 200) {

SHOWERROR(@"请输入200字以内的班级简介");

return;

}

[JXHttpTool POST:@"classIntroductionUpdate" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inIntroduction":self.textView.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

if (self.isClassessPublicNotice) {

//班级公告

kJudgeNullWith(self.topTextField.text, @"请输入公告标题");

kJudgeNullWith(self.textView.text, @"请输入公告详情");

if (self.textView.text.length < 15

|| self.textView.text.length > 500) {

SHOWERROR(@"请输入15-500字的班级公告");

return;

}

NSLog(@"%@",@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inTitle":self.topTextField.text?:EMPTY\_STRING,@"inContent":self.textView.text?:EMPTY\_STRING});

[JXHttpTool POST:@"classNoticeAdd" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inTitle":self.topTextField.text?:EMPTY\_STRING,@"inContent":self.textView.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

if (self.isEditCollect) {

//编辑收藏夹

kJudgeNullWith(self.textView.text, @"请输入收藏夹名称");

if (self.textView.text.length > 10) {

SHOWERROR(@"请输入10字以内的收藏夹名称");

return;

}

[JXHttpTool POST:@"updateFavorites" parameters:@{@"inPlatformId":self.inPlatformId?:EMPTY\_STRING,@"inId":self.collectID?:EMPTY\_STRING,@"inFavoritesName":self.textView.text?:EMPTY\_STRING} success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

if (self.isAddCollect) {

//创建收藏夹

kJudgeNullWith(self.textView.text, @"请输入收藏夹名称");

if (self.textView.text.length > 10) {

SHOWERROR(@"请输入10字以内的收藏夹名称");

return;

}

#warning 不确定inPlatformId是不是个人的

[JXHttpTool POST:@"createFavorites" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inFavoritesName":self.textView.text?:EMPTY\_STRING} success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

if (self.isAddRecord) {

//添加笔记

kJudgeNullWith(self.textView.text, @"请填写笔记内容");

NSMutableDictionary \*para = @{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inTestId":self.inTestId?:EMPTY\_STRING,@"inNotes":self.textView.text?:EMPTY\_STRING,@"inImg":\_iconDataArr?:@[].mutableCopy,@"Token":KACCOUNTTOOLTOKEN}.mutableCopy;

[JXHttpTool upLoadAddRecordWithUrlString:@"addNotes" parameters:para completionHandler:^(id responseObject, NSError \*error) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

}];

}

if (self.isModifyInfo) {

kJudgeNullWith(self.textView.text, @"请输入介绍");

if (self.textView.text.length > 50) {

SHOWERROR(@"请输入50字以内的介绍");

return;

}

[self.para setValue:self.textView.text?:EMPTY\_STRING forKey:@"inIntroduce"];

[JXHttpTool POST:@"modifyUser" parameters:self.para success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

!self.modifyIntroInfoBlock ?: self.modifyIntroInfoBlock(self.textView.text?:EMPTY\_STRING);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

}

/\*

#pragma mark - Navigation

// In a storyboard-based application, you will often want to do a little preparation before navigation

- (void)prepareForSegue:(UIStoryboardSegue \*)segue sender:(id)sender {

// Get the new view controller using [segue destinationViewController].

// Pass the selected object to the new view controller.

}

\*/

@end

//

// MineTVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/9.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineTVC.h"

#import "MineHeader.h"

#import "BDImagePicker.h"

#import "UIImage+Gaussina.h"

#import "MineClassesManager.h"

#import "MineBandScroolVC.h"

#import "MineStudentManagerTVC.h"

#import "UIImage+ImageEffects.h"

#import "MineMyInfoTVC.h"

#import "MineInfoEditTVC.h"

@interface MineTVC ()

{

NSInteger \_imageHeight;//header高度

CGFloat \_myClassHeight;

}

@property (nonatomic,strong) MineHeader \*header;

@property (weak, nonatomic) IBOutlet UILabel \*faculty;

@property (weak, nonatomic) IBOutlet UILabel \*changeSchool;

@end

@implementation MineTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

\_myClassHeight = 80;

self.automaticallyAdjustsScrollViewInsets = NO;

self.tableView.backgroundColor = OVERALLCOLOR;

self.navigationItem.leftBarButtonItem = [UIBarButtonItem barButtonItemWithTitle:@"我的" titleColor:[UIColor blackColor] andBarButtonItemBlock:^{

}];

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"icon\_edit\_Mine" andBarButtonItemBlock:^{

MineInfoEditTVC \*info = kStoryboardVC(@"Mine", @"MineInfoEditTVCSBID");

info.hasHeader = YES;

info.refreshIconBlock = ^{

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

[self refreshIcon];

}];

};

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

}];

[DefaultCenter addObserver:self selector:@selector(refreshUI) name:AfterLoginRefresh object:nil];

[self refreshUI];

[self refreshIcon];

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

[self refreshUI];

[self refreshIcon];

}];

}

-(MineHeader \*)header

{

if (!\_header) {

\_imageHeight = 200;

\_header = [MineHeader viewFromNIB];

\_header.username.text = KACCOUNTTOOLOUTUSER.User\_Name;

\_header.intro.text = KACCOUNTTOOLOUTUSER.Introduce;

WEAKSELF

\_header.clickIconBlock = ^{

[BDImagePicker showImagePickerFromViewController:weakSelf\_SC allowsEditing:YES finishAction:^(UIImage \*image) {

if (image) {

NSData \*imageData = kImageDataCompress(image);

[JXHttpTool upLoadDataWithUrlString:@"updatePhoto" parameters:imageData completionHandler:^(id responseObject, NSError \*error) {

NSLog(@"%@",responseObject);

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

[weakSelf\_SC refreshIcon];

}];

}];

}

}];

};

\_header.height = \_imageHeight;

}

return \_header;

}

- (IBAction)clickLogout:(UIButton \*)sender {

//退出登录

[KACCOUNTTOOL afterLogOut];

[self.tableView scrollToTop];

}

- (void)refreshUI

{

self.faculty.text = KACCOUNTTOOL.Faculty;

self.header.username.text = KACCOUNTTOOLOUTUSER.User\_Name;

self.header.intro.text = KACCOUNTTOOLOUTUSER.Introduce;

self.changeSchool.text = [KACCOUNTTOOLOUTSCHOOLINFO isKindOfClass:[NSNull class]]?EMPTY\_STRING:KACCOUNTTOOLOUTSCHOOLINFO.SchoolName;

}

- (void)refreshIcon

{

[self.header.icon sd\_setImageWithURL:URLWITHSTRING(KACCOUNTTOOLOUTUSER.PhotoUrl?:EMPTY\_STRING) forState:UIControlStateNormal placeholderImage:self.header.icon.currentImage completed:^(UIImage \* \_Nullable image, NSError \* \_Nullable error, SDImageCacheType cacheType, NSURL \* \_Nullable imageURL) {

self.header.bgImage.image = [image applyDarkEffect];

}];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

[self.tableView scrollToTop];

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

[self refreshUI];

if ([KACCOUNTTOOLOUTUSER.CurrentRole isEqualToString:@"0"]) {

//私人 我的班级 高度为0

\_myClassHeight = 0.00001f;

} else {

\_myClassHeight = 80;

}

[self.tableView reloadData];

}];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

- (void)scrollViewDidScroll:(UIScrollView \*)scrollView

{

CGFloat yOffset = scrollView.contentOffset.y;// 偏移的y值

CGFloat width = self.view.frame.size.width;// 图片的宽度

// NSLog(@"%f",yOffset);

if (yOffset < 0) {

CGFloat totalOffset = \_imageHeight + ABS(yOffset);

CGFloat f = totalOffset / \_imageHeight;

self.header.bgImage.frame = CGRectMake(- (width \* f - width) / 2, yOffset, width \* f, totalOffset);// 拉伸后的图片的frame应该是同比例缩放

}

}

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

{

return [super tableView:tableView cellForRowAtIndexPath:indexPath];

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

// //班级管理

// MineClassesManager \*manager = kStoryboardVC(@"Mine", @"MineClassesManagerSBID");

// manager.showTitle = @"班级管理";

// //显示选择班级

// manager.chooseClasses = YES;

// manager.SchoolModel = KACCOUNTTOOLOUTSCHOOLINFO;

// [self.navigationController pushViewController:manager animated:YES];

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

// //学生管理

// MineStudentManagerTVC \*manager = kStoryboardVC(@"Mine", @"MineStudentManagerTVCSBID");

// manager.studentManager = YES;

// manager.SchoolId = [KACCOUNTTOOLOUTSCHOOLINFO SchoolId];

// manager.SchoolNameStr = [KACCOUNTTOOLOUTSCHOOLINFO SchoolName];

// [self.navigationController pushViewController:manager animated:YES];

if (indexPath.row == 0) {

//绑定学校

MineBandScroolVC \*bandVC = kStoryboardVC(@"Mine", @"MineBandScroolVCSBID");

bandVC.isLoginComeIn = NO;

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

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

//切换学校

MineClassesManager \*manager = kStoryboardVC(@"Mine", @"MineClassesManagerSBID");

manager.showTitle = @"切换学校";

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

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

//我的班级

MineClassesManager \*manager = kStoryboardVC(@"Mine", @"MineClassesManagerSBID");

manager.showTitle = @"班级管理";

//显示选择班级

manager.chooseClasses = YES;

manager.SchoolModel = KACCOUNTTOOLOUTSCHOOLINFO;

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

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

//切换学部

[JXShowSelectTool showSelectWithArr:@[@"小学",@"初中",@"高中"] andDefaultStr:@"小学" complete:^(NSString \*selectedStr, NSInteger num) {

KACCOUNTTOOL.Faculty = selectedStr;

[KACCOUNTTOOL saveUserAccount];

[DefaultCenter postNotificationName:kChangeFacultyNofity object:nil];

self.faculty.text = KACCOUNTTOOL.Faculty;

}];

}

}

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

{

if (indexPath.row == 1) {

return \_myClassHeight;

}

return [super tableView:tableView heightForRowAtIndexPath:indexPath];

}

-(UIView \*)tableView:(UITableView \*)tableView viewForHeaderInSection:(NSInteger)section

{

return self.header;

}

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

{

return \_imageHeight + ABS(tableView.contentOffset.y) ;// 偏移的y值

}

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

{

return 0.0001f;

}

@end

//

// MineMyInfoTVC.m

// MaestroEdu

//

// Created by 张明辉 on 2017/7/10.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineMyInfoTVC.h"

#import "MineHeader.h"

#import "BDImagePicker.h"

#import "UIImage+Gaussina.h"

@interface MineMyInfoTVC ()

{

NSInteger \_imageHeight;//header高度

}

@property (nonatomic,strong) MineHeader \*header;

@property (weak, nonatomic) IBOutlet UILabel \*sex;

@property (weak, nonatomic) IBOutlet UILabel \*phoe;

@property (weak, nonatomic) IBOutlet UILabel \*platformID;

@property (weak, nonatomic) IBOutlet UILabel \*staffID;

@end

@implementation MineMyInfoTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.automaticallyAdjustsScrollViewInsets = NO;

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"icon\_edit\_Mine" andBarButtonItemBlock:^{

kPushStoryboardVC(@"Mine", @"MineInfoEditTVCSBID")

}];

\_imageHeight = 200;

MineHeader \*headerView = [MineHeader viewFromNIB];

headerView.clickIconBlock = ^{

[BDImagePicker showImagePickerFromViewController:self allowsEditing:YES finishAction:^(UIImage \*image) {

if (image) {

NSData \*imageData = kImageDataCompress(image);

[JXHttpTool upLoadDataWithUrlString:@"updatePhoto" parameters:imageData completionHandler:^(id responseObject, NSError \*error) {

NSLog(@"%@",responseObject);

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self refreshUI];

!self.refreshIcon ?: self.refreshIcon();

}];

}];

}

}];

};

headerView.height = \_imageHeight;

self.header = headerView;

[self refreshUI];

}

- (void)refreshUI

{

self.sex.text = [DJUtils getSexWithSexNum:KACCOUNTTOOLOUTUSER.Sex];

self.phoe.text = KACCOUNTTOOLOUTUSER.Phone;

self.platformID.text = KACCOUNTTOOLOUTUSER.PlatformNumber;

self.staffID.text = KACCOUNTTOOLOUTUSER.Number;

[self.header.icon sd\_setImageWithURL:URLWITHSTRING(KACCOUNTTOOLOUTUSER.PhotoUrl) forState:UIControlStateNormal placeholderImage:IMG(@"default\_avatar") completed:^(UIImage \* \_Nullable image, NSError \* \_Nullable error, SDImageCacheType cacheType, NSURL \* \_Nullable imageURL) {

self.header.bgImage.image = [image convertToBlurImage];

}];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

NSLog(@"%@",responseObject);

[self refreshUI];

}];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

- (void)scrollViewDidScroll:(UIScrollView \*)scrollView

{

CGFloat yOffset = scrollView.contentOffset.y;// 偏移的y值

CGFloat width = self.view.frame.size.width;// 图片的宽度

NSLog(@"%f",yOffset);

if (yOffset < 0) {

CGFloat totalOffset = \_imageHeight + ABS(yOffset);

CGFloat f = totalOffset / \_imageHeight;

self.header.bgImage.frame = CGRectMake(- (width \* f - width) / 2, yOffset, width \* f, totalOffset);// 拉伸后的图片的frame应该是同比例缩放

}

}

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

{

return [super tableView:tableView cellForRowAtIndexPath:indexPath];

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if (indexPath.row == 1) {

kPresentWith(@"Mine", @"MineModifyPhoneTVCSBID");

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

kPresentWith(@"Mine", @"MineModifyPwdTVCSBID");

}

}

-(UIView \*)tableView:(UITableView \*)tableView viewForHeaderInSection:(NSInteger)section

{

return self.header;

}

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

{

return \_imageHeight + ABS(tableView.contentOffset.y) ;// 偏移的y值

}

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

{

return 0.0001f;

}

@end

//

// MineInfoEditTVC.m

// MaestroEdu

//

// Created by 张明辉 on 2017/7/10.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineInfoEditTVC.h"

#import "JXShowSelectTool.h"

#import "MineHeader.h"

#import "BDImagePicker.h"

#import "UIImage+ImageEffects.h"

#import "HomeCollectCreateAndEdit.h"

#define kgbkEncoding (CFStringConvertEncodingToNSStringEncoding(kCFStringEncodingGB\_18030\_2000))

@interface MineInfoEditTVC ()

{

NSInteger \_imageHeight;//header高度

}

@property (weak, nonatomic) IBOutlet UITextField \*name;

@property (weak, nonatomic) IBOutlet UITextField \*sign;

@property (weak, nonatomic) IBOutlet UITextField \*sex;

@property (weak, nonatomic) IBOutlet UITextField \*phone;

@property (weak, nonatomic) IBOutlet UITextField \*platformID;

@property (weak, nonatomic) IBOutlet UITextField \*staffID;

@property (nonatomic,strong) MineHeader \*header;

@end

@implementation MineInfoEditTVC

- (void)viewDidLoad {

[super viewDidLoad];

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"general\_icon\_right" andBarButtonItemBlock:^{

[self commitInfoData];

}];

self.name.text = KACCOUNTTOOLOUTUSER.User\_Name;

self.sign.text = KACCOUNTTOOLOUTUSER.Introduce;

self.sex.text = [DJUtils getSexWithSexNum:KACCOUNTTOOLOUTUSER.Sex];

self.phone.text = KACCOUNTTOOLOUTUSER.Phone;

self.platformID.text = KACCOUNTTOOLOUTUSER.PlatformNumber;

#warning 学籍号

self.staffID.text = KACCOUNTTOOLOUTUSER.Number;

if (self.hasHeader) {

\_imageHeight = 200;

MineHeader \*headerView = [MineHeader viewFromNIB];

headerView.clickIconBlock = ^{

[BDImagePicker showImagePickerFromViewController:self allowsEditing:YES finishAction:^(UIImage \*image) {

if (image) {

NSData \*imageData = kImageDataCompress(image);

[JXHttpTool upLoadDataWithUrlString:@"updatePhoto" parameters:imageData completionHandler:^(id responseObject, NSError \*error) {

NSLog(@"%@",responseObject);

[JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

!self.refreshIconBlock ?: self.refreshIconBlock();

[self.header.icon sd\_setImageWithURL:URLWITHSTRING(KACCOUNTTOOLOUTUSER.PhotoUrl?:EMPTY\_STRING) forState:UIControlStateNormal placeholderImage:self.header.icon.currentImage completed:^(UIImage \* \_Nullable image, NSError \* \_Nullable error, SDImageCacheType cacheType, NSURL \* \_Nullable imageURL) {

self.header.bgImage.image = [image applyDarkEffect];

}];

}];

}];

}

}];

};

headerView.height = \_imageHeight;

self.header = headerView;

}

self.header.username.text = KACCOUNTTOOLOUTUSER.User\_Name;

[self.header.icon sd\_setImageWithURL:URLWITHSTRING(KACCOUNTTOOLOUTUSER.PhotoUrl?:EMPTY\_STRING) forState:UIControlStateNormal placeholderImage:self.header.icon.currentImage completed:^(UIImage \* \_Nullable image, NSError \* \_Nullable error, SDImageCacheType cacheType, NSURL \* \_Nullable imageURL) {

self.header.bgImage.image = [image applyDarkEffect];

}];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

if (self.hasHeader) {

self.automaticallyAdjustsScrollViewInsets = NO;

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

}

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

if (self.hasHeader) {

self.navigationController.navigationBar.translucent = NO;

}

}

- (void)commitInfoData

{

if ([self.name.text dataUsingEncoding:kgbkEncoding].length > 14) {

SHOWERROR(@"输入14位英文字符或7个汉字的姓名");

return;

}

[JXHttpTool POST:@"modifyUser" parameters:@{@"Token":KACCOUNTTOOLTOKEN,@"inNumber":KACCOUNTTOOLOUTUSER.PlatformNumber?:EMPTY\_STRING,@"inSex":[DJUtils getSexNumWithSex:self.sex.text],@"inIntroduce":self.sign.text?:EMPTY\_STRING,@"inUser\_Name":self.name.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if (indexPath.row == 1) {

HomeCollectCreateAndEdit \*classIntro = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

classIntro.isModifyInfo = YES;

classIntro.para = @{@"Token":KACCOUNTTOOLTOKEN,@"inNumber":KACCOUNTTOOLOUTUSER.PlatformNumber?:EMPTY\_STRING,@"inSex":[DJUtils getSexNumWithSex:self.sex.text],@"inIntroduce":self.sign.text?:EMPTY\_STRING,@"inUser\_Name":self.name.text?:EMPTY\_STRING}.mutableCopy;

classIntro.modifyIntroInfoBlock = ^(NSString \*intro) {

self.sign.text = intro;

};

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

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

[JXShowSelectTool showSelectWithArr:@[@"男",@"女"] andDefaultStr:@"男" complete:^(NSString \*selectedStr, NSInteger num) {

self.sex.text = selectedStr;

}];

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

kPresentWith(@"Mine", @"MineModifyPhoneTVCSBID");

} else if (indexPath.row == 6) {

kPresentWith(@"Mine", @"MineModifyPwdTVCSBID");

}

}

-(UIView \*)tableView:(UITableView \*)tableView viewForHeaderInSection:(NSInteger)section

{

return self.hasHeader? self.header : nil;

}

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

{

return self.hasHeader?( \_imageHeight + ABS(tableView.contentOffset.y)) : 0.00001f ;// 偏移的y值

}

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

{

return 0.0001f;

}

- (void)scrollViewDidScroll:(UIScrollView \*)scrollView

{

CGFloat yOffset = scrollView.contentOffset.y;// 偏移的y值

CGFloat width = self.view.frame.size.width;// 图片的宽度

NSLog(@"%f",yOffset);

if (yOffset < 0) {

CGFloat totalOffset = \_imageHeight + ABS(yOffset);

CGFloat f = totalOffset / \_imageHeight;

self.header.bgImage.frame = CGRectMake(- (width \* f - width) / 2, yOffset, width \* f, totalOffset);// 拉伸后的图片的frame应该是同比例缩放

}

}

@end

//

// MineStudentManagerTVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/10.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineStudentManagerTVC.h"

#import "MineStudentManagerModel.h"

#import "MineStudentManageEditTVC.h"

#import "MineTeacherManagerCell.h"

#import "FTPopOverMenu.h"

@interface MineStudentManagerTVC ()

{

NSString \*\_inType;

}

@property (weak, nonatomic) IBOutlet UIView \*searchBgView;

@property (weak, nonatomic) IBOutlet UITextField \*input;

@property (weak, nonatomic) IBOutlet UILabel \*schoolName;

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

@end

@implementation MineStudentManagerTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

if (self.showTitle) {

self.title = self.showTitle;

}

//如果从班级详情进入查询学生列表，则不显示

if (self.addStudent) {

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"screening" andBarButtonItemBlock:^{

[self showTypeList];

}];

}

if (self.studentManager) {

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"icon\_add" andBarButtonItemBlock:^{

MineStudentManageEditTVC \*editVC = kStoryboardVC(@"Mine", @"MineStudentManageEditTVCSBID");

editVC.SchoolId = self.SchoolId;

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

}];

}

[self.input addTarget:self action:@selector(refreshDataFliter) forControlEvents:UIControlEventEditingChanged];

self.searchBgView.layer.borderColor = BORDERCOLOR.CGColor;

self.searchBgView.layer.cornerRadius = self.searchBgView.height \* 0.5;

self.searchBgView.layer.borderWidth = 1;

self.selectedStudent = @[].mutableCopy;

if (self.className) {

//班级名称

self.schoolName.text = self.className;

} else {

//学校名称

self.schoolName.text = self.SchoolNameStr;

}

if (self.paperAttribute || self.paperAttributeShowClassess) {

//只有下拉刷新

WEAKSELF

[self.tableView addHeaderRefresh:^{

//请求所有学生列表

kTableViewHeaderAction(weakSelf\_SC.tableView);

[weakSelf\_SC refreshData];

[weakSelf\_SC.tableView endHeaderRefresh];

}];

kTableViewInit(self.tableView);

} else {

//上下拉刷新

WEAKSELF

kTableViewFuckingInit(weakSelf\_SC.tableView, refreshData);

}

}

- (void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

kTableViewHeaderAction(self.tableView);

[self refreshData];

}

- (void)showTypeList

{

NSArray \*images = @[@"全部", @"已在班级学生", @"未在班级学生"];

[FTPopOverMenu showFromSenderFrame:CGRectMake(self.view.frame.size.width - 40, 20, 40, 40)

withMenuArray:images

imageArray:nil

doneBlock:^(NSInteger selectedIndex) {

NSLog(@"done block. do something. selectedIndex : %ld", (long)selectedIndex);

\_inType = [NSString stringWithFormat:@"%ld",(long)selectedIndex];

[self requestClassStudentListData];

} dismissBlock:^{

NSLog(@"user canceled. do nothing.");

}];

}

- (void)refreshDataFliter

{

self.tableView.CurrentPage = @1;

[self.tableView resetNoMoreData];

[self.tableView.tableViewDatas removeAllObjects];

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

[self.tableView endHeaderRefresh];

[self refreshData];

}

- (void)refreshData

{

if (self.lookStudent) {

//从班级详情s进入学生列表

\_inType = @"1";

[self requestClassStudentListData];

} else if (self.addStudent) {

//从班级详情s进入学生列表

\_inType = @"2";

[self requestClassStudentListData];

} else if (self.paperAttribute) {

//从试卷配置里进入学生列表

[self requestPaperAttributeSelectStudent];

} else if (self.paperAttributeShowClassess){

//从试卷配置里进入班级列表

[self requestPaperAttributeSelectClassess];

} else {

//请求所有学生列表

[self requestDataAll];

}

}

- (void)requestPaperAttributeSelectStudent

{

[JXHttpTool POST:@"selTestStudent" parameters:@{} success:^(id responseObject) {

self.tableView.tableViewDatas = [NSMutableArray arrayWithArray:responseObject[@"outList"]];

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:nil];

}

- (void)requestPaperAttributeSelectClassess

{

[JXHttpTool POST:@"myClassList" parameters:@{@"inId":KACCOUNTTOOLOUTUSER.CurrentRole?:EMPTY\_STRING,@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber} success:^(id responseObject) {

self.tableView.tableViewDatas = [NSMutableArray arrayWithArray:responseObject[@"outList"]];

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)requestDataAll {

//,@"incurrentPage":@"1",@"inshowCount":@"20"

[JXHttpTool POST:@"studentList" parameters:@{@"inSchoolId":self.SchoolId?:EMPTY\_STRING,@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"keywords":self.input.text?:EMPTY\_STRING,@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"20"} success:^(id responseObject) {

[self.tableView.tableViewDatas addObjectsFromArray:[MineStudentManagerModel parse: responseObject[@"outList"]]];

if (responseObject[@"Page"] && [responseObject[@"Page"][@"CurrentPage"] isEqualToNumber:responseObject[@"Page"][@"TotalPage"]]) {

[self.tableView endFooterRefreshWithNoMoreData];

} else {

self.tableView.CurrentPage = @([self.tableView.CurrentPage intValue] + 1);

}

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)requestClassStudentListData

{

NSDictionary \*para = @{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inSchoolId":self.SchoolId?:EMPTY\_STRING,@"inType":\_inType,@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"20",@"inStudentName":self.input.text?:EMPTY\_STRING};

[JXHttpTool POST:@"classStudentSel" parameters:para success:^(id responseObject) {

[self.tableView.tableViewDatas addObjectsFromArray:responseObject[@"outList"]];

if (responseObject[@"Page"] && [responseObject[@"Page"][@"CurrentPage"] isEqualToNumber:responseObject[@"Page"][@"TotalPage"]]) {

[self.tableView endFooterRefreshWithNoMoreData];

} else {

self.tableView.CurrentPage = @([self.tableView.CurrentPage intValue] + 1);

}

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)removeStudentWithID:(NSString \*)classStudentId

{

[UIAlertTool showAlertViewWithTitle:@"提示" message:@"确认将该学员移除班级吗" cancelButtonTitle:@"取消" otherButtonTitle:@"确定" confirm:^{

//移除学生

[JXHttpTool POST:@"classStudentDel" parameters:@{@"inClassStudentId":classStudentId,@"inClassId":self.inClassId?:EMPTY\_STRING} success:^(id responseObject) {

[self refreshData];

} failure:^(id error) {

NSLog(@"%@",error);

}];

} cancle:nil];

}

- (void)addStudentWithID:(NSString \*)classStudentId

{

[UIAlertTool showAlertViewWithTitle:@"提示" message:@"确认将该学员加入班级吗" cancelButtonTitle:@"取消" otherButtonTitle:@"确定" confirm:^{

//添加学生

[JXHttpTool POST:@"classStudentAdd" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inStudentId":classStudentId?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self requestClassStudentListData];

} failure:^(id error) {

NSLog(@"%@",error);

}];

} cancle:nil];

}

-(void)clickBottomBtn

{

//组卷进入

!self.clickFinishSelect ?: self.clickFinishSelect(self.selectedStudent);

[self.navigationController popViewControllerAnimated:YES];

}

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

{

if (section == 0) {

return 2;

}

return self.tableView.tableViewDatas.count;

}

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

{

if (indexPath.section == 0) {

//学校、搜索 用Static cell

return [super tableView:tableView cellForRowAtIndexPath:indexPath];

}

MineTeacherManagerCell \*cell = [MineTeacherManagerCell cellWithNibTableView:tableView andReuseID:@"MineTeacherManagerCellSBID"];

cell.teacherName.hidden = YES;

cell.rightBtn.enabled = NO;

cell.rightBtn.userInteractionEnabled = NO;

cell.rightBtn.hidden = YES;

if (self.lookStudent || self.addStudent) {

//班级详情进入学生列表

cell.teacher.text = self.tableView.tableViewDatas[indexPath.row][@"StudentName"];

// cell.rightBtn.hidden = NO;

if ([self.tableView.tableViewDatas[indexPath.row][@"ClassStudentId"] isEqualToNumber:@0]) {

//未在该班级

[cell.rightBtn setImage:IMG(@"Add\_student") forState:UIControlStateNormal];

cell.clickRightBtnBlock = ^{

//添加学生

[self addStudentWithID:self.tableView.tableViewDatas[indexPath.row][@"StudentId"]];

};

} else {

//已在该班级

[cell.rightBtn setImage:IMG(@"Remove\_student") forState:UIControlStateNormal];

cell.clickRightBtnBlock = ^{

//移除学生

[self removeStudentWithID:self.tableView.tableViewDatas[indexPath.row][@"ClassStudentId"]];

};

}

} else if (self.paperAttribute){

cell.teacher.text = self.tableView.tableViewDatas[indexPath.row][@"StudentName"];

cell.rightBtn.hidden = NO;

cell.rightBtn.userInteractionEnabled = NO;

[cell.rightBtn setImage:IMG(@"Check\_box\_default") forState:UIControlStateNormal];

[cell.rightBtn setImage:IMG(@"Check\_box\_selected") forState:UIControlStateSelected];

} else if (self.paperAttributeShowClassess) {

cell.teacher.text = self.tableView.tableViewDatas[indexPath.row][@"ClassName"];

cell.rightBtn.hidden = NO;

cell.rightBtn.userInteractionEnabled = NO;

[cell.rightBtn setImage:IMG(@"Check\_box\_default") forState:UIControlStateNormal];

[cell.rightBtn setImage:IMG(@"Check\_box\_selected") forState:UIControlStateSelected];

} else {

cell.teacher.text = [self.tableView.tableViewDatas[indexPath.row] User\_Name];

cell.rightBtn.hidden = YES;

}

return cell;

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if (indexPath.section == 0) {

return;

}

if (!self.lookStudent && !self.addStudent && !self.paperAttribute && !self.paperAttributeShowClassess) {

//不是班级详情进入

MineStudentManageEditTVC \*editVC = kStoryboardVC(@"Mine", @"MineStudentManageEditTVCSBID");

editVC.SchoolId = self.SchoolId;

editVC.isEdit = YES;

editVC.model = self.tableView.tableViewDatas[indexPath.row];

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

}

if (self.paperAttribute || self.paperAttributeShowClassess) {

//组卷进入

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

cell.rightBtn.selected = !cell.rightBtn.selected;

if (cell.rightBtn.selected) {

[self.selectedStudent addObject:self.tableView.tableViewDatas[indexPath.row]];

} else {

if ([self.selectedStudent containsObject:self.tableView.tableViewDatas[indexPath.row]]) {

[self.selectedStudent removeObject:self.tableView.tableViewDatas[indexPath.row]];

}

}

}

}

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

{

if (indexPath.section == 0) {

//学校、搜索 用Static cell

return [super tableView:tableView heightForRowAtIndexPath:indexPath];

}

return 44;

}

@end

//

// MineMyClassessTVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/9.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineMyClassessTVC.h"

#import "HomeCollectCreateAndEdit.h"

#import "MineMyClassessDetailModel.h"

#import "MineChangeTeacherTVC.h"

#import "MineTeacherManagerTVC.h"

#import "MineStudentManagerTVC.h"

#import "MineMyClassessTeacherModel.h"

#import "MineHeader.h"

@interface MineMyClassessTVC ()

{

NSInteger \_imageHeight;//header高度

}

@property (nonatomic,strong) MineHeader \*header;

@property (nonatomic,strong) MineMyClassessDetailModel \*model;

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

@property (weak, nonatomic) IBOutlet UILabel \*publicNotice;

@property (weak, nonatomic) IBOutlet UILabel \*teacherClass;

@property (weak, nonatomic) IBOutlet UILabel \*topTitle;

@property (weak, nonatomic) IBOutlet UILabel \*classTitl;

@property (weak, nonatomic) IBOutlet UILabel \*teacherNum;

@property (weak, nonatomic) IBOutlet UILabel \*studentNum;

@end

@implementation MineMyClassessTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view

\_imageHeight = 200;

self.teacherArr = @[].mutableCopy;

self.automaticallyAdjustsScrollViewInsets = NO;

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

[self requestData];

}

- (IBAction)addTeacher:(UIButton \*)sender {

//教师列表

// [self goTeacherList];

}

- (IBAction)addStudent:(UIButton \*)sender {

//学生列表

// [self goStudentList:YES];

}

- (void)goTeacherList

{

MineTeacherManagerTVC \*teacherManager = kStoryboardVC(@"Mine", @"MineTeacherManagerTVCSBID");

teacherManager.classNameStr = self.model.ClassName;

teacherManager.outSubject = self.teacherArr;

teacherManager.SchoolId = self.SchoolModel.SchoolId;

teacherManager.inClassId = self.inClassId;

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

}

- (void)goStudentList:(BOOL)isAdd

{

//是不是添加学生

MineStudentManagerTVC \*studentManager = kStoryboardVC(@"Mine", @"MineStudentManagerTVCSBID");

if (isAdd) {

studentManager.addStudent = YES;

} else {

studentManager.lookStudent = YES;

}

studentManager.className = self.model.ClassName;

studentManager.inClassId = self.inClassId?:EMPTY\_STRING;

studentManager.SchoolId = self.SchoolModel.SchoolId;

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

}

-(MineHeader \*)header

{

if (!\_header) {

\_imageHeight = 200;

\_header = [MineHeader viewFromNIB];

\_header.username.text = KACCOUNTTOOLOUTUSER.User\_Name;

// WEAKSELF

\_header.clickIconBlock = ^{

// [BDImagePicker showImagePickerFromViewController:weakSelf\_SC allowsEditing:YES finishAction:^(UIImage \*image) {

// if (image) {

// NSData \*imageData = kImageDataCompress(image);

//

// [JXHttpTool upLoadDataWithUrlString:@"updatePhoto" parameters:imageData completionHandler:^(id responseObject, NSError \*error) {

// NSLog(@"%@",responseObject);

//

// [JXAccountTool refreshUserInfoSuccess:^(id responseObject) {

//

// SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

//

// [weakSelf\_SC refreshUI];

//

//

// }];

//

// }];

//

// }

// }];

};

\_header.height = \_imageHeight;

}

return \_header;

}

- (void)scrollViewDidScroll:(UIScrollView \*)scrollView

{

CGFloat yOffset = scrollView.contentOffset.y;// 偏移的y值

CGFloat width = self.view.frame.size.width;// 图片的宽度

// NSLog(@"%f",yOffset);

if (yOffset < 0) {

CGFloat totalOffset = \_imageHeight + ABS(yOffset);

CGFloat f = totalOffset / \_imageHeight;

self.header.bgImage.frame = CGRectMake(- (width \* f - width) / 2, yOffset, width \* f, totalOffset);// 拉伸后的图片的frame应该是同比例缩放

}

}

- (void)requestData

{

[JXHttpTool POST:@"classShow" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING} success:^(id responseObject) {

self.model = [MineMyClassessDetailModel parse:responseObject[@"outList"]];

self.topTitle.text = self.model.ClassName;

self.publicNotice.text = self.model.NoticeTitle;

self.teacherClass.text = self.model.ClassAdviser;

self.classTitl.text = self.model.ClassName;

self.teacherNum.text = [NSString stringWithFormat:@"%@人",self.model.TeacherNum];

self.studentNum.text = [NSString stringWithFormat:@"%@人",self.model.StudentNum];

self.teacherArr = [NSMutableArray arrayWithArray:[MineMyClassessTeacherModel parse:responseObject[@"outSubject"]]];

[self.header.icon sd\_setImageWithURL:URLWITHSTRING(KACCOUNTTOOLOUTUSER.PhotoUrl?:EMPTY\_STRING) forState:UIControlStateNormal placeholderImage:IMG(@"default\_avatar") completed:^(UIImage \* \_Nullable image, NSError \* \_Nullable error, SDImageCacheType cacheType, NSURL \* \_Nullable imageURL) {

// self.header.bgImage.image = [image applyDarkEffect];

}];

[self.tableView reloadData];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if (indexPath.row == 1) {

//简介

HomeCollectCreateAndEdit \*classIntro = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

classIntro.isClassessIntro = YES;

classIntro.inClassId = self.inClassId;

classIntro.classessIntroStr = self.model.Introduction;

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

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

//公告

HomeCollectCreateAndEdit \*classIntro = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

classIntro.isClassessPublicNotice = YES;

classIntro.inClassId = self.inClassId;

classIntro.publicNoticeTitle = self.model.NoticeTitle;

classIntro.publicNoticeDetail = self.model.NoticeContent;

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

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

// //更换班主任

// MineChangeTeacherTVC \*changeTeacherTVC = kStoryboardVC(@"Mine",@"MineChangeTeacherTVCSBID");

// changeTeacherTVC.schoolId = self.SchoolModel.SchoolId;

// changeTeacherTVC.inClassId = self.inClassId;

// [self.navigationController pushViewController:changeTeacherTVC animated:YES];

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

//教师列表

[self goTeacherList];

} else if (indexPath.row == 7) {

//学生列表

[self goStudentList:NO];

}

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

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

{

if (

indexPath.row == 0

|| indexPath.row == 1

|| indexPath.row == 2

|| indexPath.row == 9) {

return 0.00001f;

}

return [super tableView:tableView heightForRowAtIndexPath:indexPath];

}

-(UIView \*)tableView:(UITableView \*)tableView viewForHeaderInSection:(NSInteger)section

{

return self.header;

}

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

{

return \_imageHeight + ABS(tableView.contentOffset.y) ;// 偏移的y值

}

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

{

return 0.0001f;

}

@end

//

// MineChangeTeacherTVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/10.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineChangeTeacherTVC.h"

#import "MineChangeTeacherModel.h"

#import "MineMyClassessTeacherModel.h"

@interface MineChangeTeacherTVC ()

@property (weak, nonatomic) IBOutlet UIView \*searchBgView;

@property (weak, nonatomic) IBOutlet UITextField \*input;

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

@end

@implementation MineChangeTeacherTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

// self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"general\_icon\_right" andBarButtonItemBlock:^{

//

// }];

self.datas = @[].mutableCopy;

if (self.showTitle) {

self.title = self.showTitle;

}

[self.input addTarget:self action:@selector(requestData) forControlEvents:UIControlEventEditingChanged];

self.searchBgView.layer.borderColor = BORDERCOLOR.CGColor;

self.searchBgView.layer.cornerRadius = self.searchBgView.height \* 0.5;

self.searchBgView.layer.borderWidth = 1;

if (self.selectTeacherModel) {

//选择科目对应的教师

[self requestSubjectTeacher];

} else {

//查询学校的所有教师 更换班主任

[self requestData];

}

}

- (void)requestSubjectTeacher

{

[JXHttpTool POST:@"schoolTeacher" parameters:@{@"inSchoolId":self.schoolId?:EMPTY\_STRING,@"inSubjectId":self.selectTeacherModel.SubjectId?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

self.datas = [MineChangeTeacherModel parse: responseObject[@"outList"]];

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)addClassTeacherWith:(MineChangeTeacherModel \*)model

{

[JXHttpTool POST:@"addClassTeacher" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inTeacherId":model.TeacherId?:EMPTY\_STRING,@"inSubjectsId":model.SubjectId?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)modifyClassTeacherWith:(MineChangeTeacherModel \*)model

{

[JXHttpTool POST:@"updateClassTeacher" parameters:@{@"inClassId":self.inClassId?:EMPTY\_STRING,@"inTeacherId":model.TeacherId?:EMPTY\_STRING,@"inSubjectsId":model.SubjectId?:EMPTY\_STRING,@"inClassTeacherId":self.selectTeacherModel.ClassTeacherId?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)requestData {

[JXHttpTool POST:@"mySchoolTeacherAll" parameters:@{@"inSchoolId":self.schoolId?:EMPTY\_STRING,@"inTeacherName":self.input.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

self.datas = [MineChangeTeacherModel parse: responseObject[@"outList"]];

[self.tableView reloadSection:1 withRowAnimation:UITableViewRowAnimationFade];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (void)changeTeacherWith:(MineChangeTeacherModel \*)teacherModel

{

[JXHttpTool POST:@"classAdviserUpdate" parameters:@{@"inClassAdviserId":teacherModel.TeacherId?:EMPTY\_STRING,@"inId":self.inClassId?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

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

{

if (section == 0) {

return 1;

}

return self.datas.count;

}

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

{

if (indexPath.section == 0) {

//学校、搜索 用Static cell

return [super tableView:tableView cellForRowAtIndexPath:indexPath];

}

MineMessageCell \*cell = [MineMessageCell cellWithNibTableView:tableView andReuseID:@"MineStudentManagerTVCCell"];

cell.showTitle.text = [self.datas[indexPath.row] TeacherName];

return cell;

}

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

{

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

if (indexPath.section == 0) {

return;

}

if (self.selectTeacherModel) {

//更换对应科组的老师

if ([self.selectTeacherModel.ClassTeacherId isEqualToString:EMPTY\_STRING]) {

//添加老师

[self addClassTeacherWith:self.datas[indexPath.row]];

} else {

//修改老师

[self modifyClassTeacherWith:self.datas[indexPath.row]];

}

} else {

//切换班主任

[self changeTeacherWith:self.datas[indexPath.row]];

}

}

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

{

if (indexPath.section == 0) {

//学校、搜索 用Static cell

return [super tableView:tableView heightForRowAtIndexPath:indexPath];

}

return 44;

}

@end

//

// MineBandScroolVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/9.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "MineBandScroolVC.h"

#import "PYSearch.h"

#import "MineScroolListTVC.h"

@interface MineBandScroolVC ()<PYSearchViewControllerDelegate,PYSearchViewControllerDataSource>

{

PYSearchViewController \*\_searchViewController;

}

@property (weak, nonatomic) IBOutlet UIView \*scroolBgView;

@property (nonatomic,strong) MineScroolListTVC \*listTVC;

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

@property (weak, nonatomic) IBOutlet UITextField \*schoolName;

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

@property (weak, nonatomic) IBOutlet UITextField \*idCard;

@property (weak, nonatomic) IBOutlet UIButton \*cancelBandBtn;

@end

@implementation MineBandScroolVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

[self.scroolBgView setTapActionWithBlock:^{

[self pushSearchVC];

}];

self.listTVC = [[MineScroolListTVC alloc] initWithStyle:UITableViewStylePlain];

// self.idCard.text = @"110000197812068973";

if (self.isLoginComeIn) {

self.cancelBandBtn.hidden = NO;

} else {

self.cancelBandBtn.hidden = YES;

}

self.schoolName.textColor = [UIColor whiteColor];

self.idCard.textColor = [UIColor whiteColor];

[self.schoolName setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

[self.idCard setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

}

- (void)pushSearchVC

{

PYSearchViewController \*searchViewController = [PYSearchViewController searchViewControllerWithHotSearches:nil searchBarPlaceholder:@"输入学校名字" didSearchBlock:^(PYSearchViewController \*searchViewController, UISearchBar \*searchBar, NSString \*searchText) {

// 开始搜索执行以下代码

// 如：跳转到指定控制器

NSLog(@"%@", searchText);

// [self.listTVC refreshSchoolListWith:searchText];

}];

// searchViewController.searchResultShowMode = PYSearchResultShowModeEmbed;

[searchViewController.searchSuggestionView setOrigin:CGPointMake(0, 0)];

searchViewController.delegate = self;

searchViewController.dataSource = self;

// searchViewController.searchResultController = self.listTVC;

searchViewController.showHotSearch = NO;

searchViewController.showSearchHistory = NO;

searchViewController.showSearchResultWhenSearchBarRefocused = YES;

\_searchViewController = searchViewController;

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

[self searchSchoolDataWithSearchText:nil];

}

- (IBAction)clickBand:(UIButton \*)sender {

kJudgeNullWith(self.schoolName.text, @"请选择学校");

kJudgeNullWith(self.idCard.text, @"请输入身份证号");

kJudgeIDCard(self.idCard.text);

[JXHttpTool POST:sender urlString:@"boundSschool" parameters:@{@"inSchool":self.schoolID?:EMPTY\_STRING,@"inIdCard":self.idCard.text?:EMPTY\_STRING,@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[KACCOUNTTOOL afterLogOut];

[self.navigationController popToRootViewControllerAnimated:NO];

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (IBAction)cancelBand:(UIButton \*)sender {

[UserDefault setObject:@"1" forKey:KSkipBandSchool];

[[JXAccountTool sharedTool] afterLogin];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

- (void)searchSchoolDataWithSearchText:(NSString \*)searchText

{

[self.resultDatas removeAllObjects];

\_searchViewController.searchSuggestions = nil;

NSDictionary \*dict;

if (!searchText || [searchText isEqualToString:EMPTY\_STRING]) {

dict = nil;

} else {

dict = @{@"inSchoolName":searchText?:EMPTY\_STRING};

}

[JXHttpTool POST:@"schoolInfoList" parameters:dict success:^(id responseObject) {

NSLog(@"%@",responseObject);

self.resultDatas = [NSMutableArray arrayWithArray: responseObject[@"SchoolInfos"]];

NSMutableArray \*arr = @[].mutableCopy;

for (NSDictionary \*dict in self.resultDatas) {

[arr addObject:dict[@"SchoolName"]];

}

\_searchViewController.searchSuggestions = arr;

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

#pragma mark - PYSearchViewControllerDelegate

- (void)searchViewController:(PYSearchViewController \*)searchViewController searchTextDidChange:(UISearchBar \*)seachBar searchText:(NSString \*)searchText

{

[self searchSchoolDataWithSearchText:searchText];

}

-(void)searchViewController:(PYSearchViewController \*)searchViewController didSelectSearchSuggestionAtIndexPath:(NSIndexPath \*)indexPath searchBar:(UISearchBar \*)searchBar

{

self.schoolName.text = \_searchViewController.searchSuggestions[indexPath.row];

self.schoolID = self.resultDatas[indexPath.row][@"SchoolId"];

[self.navigationController popViewControllerAnimated:YES];

}

-(void)didClickCancel:(PYSearchViewController \*)searchViewController

{

[self.navigationController popViewControllerAnimated:YES];

}

@end

//

// RecordStuTVC.m

// MaestroEdu

//

// Created by jesse on 2017/9/5.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "RecordStuTVC.h"

#import "HomeTopView.h"

#import "FTPopOverMenu.h"

#import "JXTimeLineModel.h"

#import "JXTimeLineCell.h"

#import "PaperBrowseVC.h"

#import "PaperListFinashTVC.h"

@interface RecordStuTVC ()

{

NSMutableArray \*\_subjectArrs;

NSString \*\_overedNumStr;

NSString \*\_collectedNumStr;

}

@end

@implementation RecordStuTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

\_subjectArrs = @[].mutableCopy;

\_overedNumStr = @"0";

\_collectedNumStr = @"0";

self.view.backgroundColor = OVERALLCOLOR;

self.automaticallyAdjustsScrollViewInsets = NO;

self.navigationItem.leftBarButtonItem = [UIBarButtonItem barButtonItemWithTitle:@"记录" titleColor:[UIColor whiteColor] andBarButtonItemBlock:^{

}];

//请求学科信息

[self requestSubjectData];

WEAKSELF

kTableViewFuckingInit(weakSelf\_SC.tableView, requestData);

[self requestData];

}

- (void)requestData

{

[JXHttpTool POST:@"testPaperTasklist" parameters:@{@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"20",@"inSubjectName":KACCOUNTTOOL.subjectImg?:EMPTY\_STRING} success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

SHOWERRORAFTERONESECOND(@"暂无数据");

kTableViewHeaderAction(self.tableView);

return ;

}

NSMutableArray \*temp = @[].mutableCopy;

for (int i = 0; i < [responseObject[@"outList"] count]; i ++) {

JXTimeLineModel \*model = [JXTimeLineModel modelWith:@{@"leftTitle":responseObject[@"outList"][i][@"ReleaseDate"],@"leftTitle2":responseObject[@"outList"][i][@"Title"],@"rightTitle1":responseObject[@"outList"][i][@"Title"],@"rightTitle2":responseObject[@"outList"][i][@"CorrectRate"],@"status":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"Isstate"]],@"TestPaperId":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"TestPaperId"]],@"StudentId":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"StudentId"]],@"ClassId":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"ClassId"]],@"SchoolId":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"SchoolId"]],@"UseTime":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"UseTime"]?:EMPTY\_STRING],@"User\_Name":[NSString stringWithFormat:@"%@",responseObject[@"outList"][i][@"User\_Name"]?:EMPTY\_STRING]}];

[temp addObject:model];

}

kTableViewAfterDataAction(self.tableView, responseObject, temp);

} failure:nil];

}

- (void)requestSubjectData

{

[JXHttpTool POST:@"subjectList" parameters:@{} success:^(id responseObject) {

NSMutableArray \*tempArr1 = [NSMutableArray arrayWithArray:responseObject[@"outList"]];

NSMutableArray \*tempArr = @[].mutableCopy;

for (NSDictionary \*dict in tempArr1) {

if ([dict[@"Faculty"] containsString:@"部"]) {

NSString \*temp = [dict[@"Faculty"] stringByReplacingOccurrencesOfString:@"部" withString:EMPTY\_STRING];

if ([temp isEqualToString:KACCOUNTTOOL.Faculty]) {

[tempArr addObject:dict];

}

}

}

\_subjectArrs = tempArr;

[self showSubjectList];

} failure:nil];

}

- (void)showSubjectList

{

NSMutableArray \*tempArr = @[].mutableCopy;

for (NSDictionary \*dict in \_subjectArrs) {

[tempArr addObject:dict[@"SubjectsName"]];

}

NSString \*showSubject;

//默认第一个

if (!KACCOUNTTOOL.subject) {

if (tempArr.count == 0) {

return;

}

showSubject = tempArr[0];

KACCOUNTTOOL.subject = [NSString stringWithFormat:@"%@",\_subjectArrs[0][@"SubjectsId"]];

KACCOUNTTOOL.subjectImg = [NSString stringWithFormat:@"%@",\_subjectArrs[0][@"SubjectsName"]];

[KACCOUNTTOOL saveUserAccount];

[self postNotificationWith:\_subjectArrs[0]];

} else {

showSubject = KACCOUNTTOOL.subjectImg;

}

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:showSubject andBarButtonItemBlock:^{

[FTPopOverMenu showFromSenderFrame:CGRectMake(self.view.frame.size.width - 40, 20, 40, 40)

withMenuArray:tempArr

imageArray:nil

doneBlock:^(NSInteger selectedIndex) {

UIButton \*btn = (UIButton \*)self.navigationItem.rightBarButtonItem.customView;

[btn setBackgroundImage:IMG(tempArr[selectedIndex]) forState:UIControlStateNormal];

[btn setBackgroundImage:IMG(tempArr[selectedIndex]) forState:UIControlStateHighlighted];

NSLog(@"done block. do something. selectedIndex : %ld", (long)selectedIndex);

//切换科目 刷新

KACCOUNTTOOL.subject = [NSString stringWithFormat:@"%@",\_subjectArrs[selectedIndex][@"SubjectsId"]];

KACCOUNTTOOL.subjectImg = [NSString stringWithFormat:@"%@",\_subjectArrs[selectedIndex][@"SubjectsName"]];

[KACCOUNTTOOL saveUserAccount];

kTableViewHeaderAction(self.tableView);

[self requestData];

[self postNotificationWith:\_subjectArrs[selectedIndex]];

} dismissBlock:^{

NSLog(@"user canceled. do nothing.");

}];

}];

[DefaultCenter addObserver:self selector:@selector(subjectRefresh) name:kChangeSubjectNofity object:nil];

}

-(void)subjectRefresh

{

//刷新logo

UIButton \*btn = self.navigationItem.rightBarButtonItem.customView;

[btn setBackgroundImage:IMG(KACCOUNTTOOL.subjectImg) forState:UIControlStateNormal];

}

- (void)postNotificationWith:(NSDictionary \*)dict

{

[DefaultCenter postNotificationName:kChangeSubjectNofity object:nil];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

if (\_subjectArrs) {

[self showSubjectList];

}

[self reqeustNumData];

kTableViewHeaderAction(self.tableView);

[self requestData];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

- (void)reqeustNumData

{

[JXHttpTool POST:@"allTestStatistics" parameters:@{} success:^(id responseObject) {

\_overedNumStr = [NSString stringWithFormat:@"%@",responseObject[@"ConquerCount"]?:EMPTY\_STRING];

\_collectedNumStr = [NSString stringWithFormat:@"%@",responseObject[@"FavoCount"]?:EMPTY\_STRING];

[self.tableView reloadData];

} failure:nil];

}

#pragma tableview datasource

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

{

return 1 + self.tableView.tableViewDatas.count;

}

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

{

if (indexPath.row == 0) {

UITableViewCell \*cell = [tableView dequeueReusableCellWithIdentifier:@"financialCell"];

if (cell == nil) {

cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"financialCell"];

HomeTopView \*topView = [HomeTopView viewFromNIB];

topView.organizeImageView.image = IMG(@"icon\_errorbank\_top");

[topView.organizeBtn setTitle:@"已攻克:0题" forState:UIControlStateNormal];

[topView.collectionBtn setTitle:@"已收藏:0题" forState:UIControlStateNormal];

topView.frame = cell.contentView.bounds;

//点击错图

topView.clickOrganizeBlock = ^{

kPushVC(@"ErrorPaperVC");

};

//点击收藏

topView.clickCollectionBlock = ^{

kPushStoryboardVC(@"Home", @"HomeCollectTVCSBID");

};

[cell.contentView addSubview:topView];

} else {

for (UIView \*view in cell.contentView.subviews) {

if ([view isKindOfClass:[HomeTopView class]]) {

HomeTopView \*topView = (HomeTopView \*)view;

[topView.organizeBtn setTitle:[NSString stringWithFormat:@"已攻克:%@题",\_overedNumStr?:@"0"] forState:UIControlStateNormal];

[topView.collectionBtn setTitle:[NSString stringWithFormat:@"已收藏:%@题",\_collectedNumStr?:@"0"] forState:UIControlStateNormal];

}

}

}

return cell;

}

JXTimeLineCell \*cell = [JXTimeLineCell cellWithNibTableView:tableView andReuseID:@"JXTimeLineCellID"];

[cell addDataWithModel:self.tableView.tableViewDatas[indexPath.row - 1]];

if (indexPath.row == 1) {

cell.topLine.hidden = YES;

if (indexPath.row == self.tableView.tableViewDatas.count) {

cell.bottomLine.hidden = YES;

}

} else if (indexPath.row == self.tableView.tableViewDatas.count) {

cell.bottomLine.hidden = YES;

} else {

cell.topLine.hidden = NO;

cell.bottomLine.hidden = NO;

}

return cell;

}

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

{

if (indexPath.row == 0) {

return;

}

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

JXTimeLineModel \*model = self.tableView.tableViewDatas[indexPath.row - 1];

if ([model.status isEqualToString:@"2"]

|| [model.status isEqualToString:@"1"]) {

//完成 批改中

PaperListFinashTVC \*finashVC = [[PaperListFinashTVC alloc] init];

finashVC.inId = @([model.inId integerValue]);

finashVC.inClassId = @([model.ClassId integerValue]);

finashVC.inStudentId = @([model.StudentId integerValue]);

if ([model.status isEqualToString:@"2"]) {

finashVC.isCorrectting = NO;

} else {

finashVC.isCorrectting = YES;

}

finashVC.showTitle = @"练习回顾";

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

} else if ([model.status isEqualToString:@"0"]) {

//去做作业

PaperBrowseVC \*paperBrowseVC = [[PaperBrowseVC alloc] init];

paperBrowseVC.HomeWorkStuComeIn = YES;

paperBrowseVC.inId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] inId]];

paperBrowseVC.StudentId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] StudentId]];

paperBrowseVC.ClassId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] ClassId]];

paperBrowseVC.SchoolId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] SchoolId]];

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

}

}

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

{

if (indexPath.row == 0) {

return 200;

}

self.tableView.rowHeight = UITableViewAutomaticDimension;

self.tableView.estimatedRowHeight = 300;

return self.tableView.rowHeight;

}

-(NSInteger)tableView:(UITableView \*)tableView indentationLevelForRowAtIndexPath:(NSIndexPath \*)indexPath

{

return 0.0001f;

}

@end

//

// ErrorPaperVC.m

// MaestroEdu

//

// Created by jesse on 2017/9/5.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "ErrorPaperVC.h"

#import "ErrorPaperSonTVC.h"

#import "FTPopOverMenu.h"

@interface ErrorPaperVC ()<VTMagicViewDataSource,VTMagicViewDelegate>

@end

@implementation ErrorPaperVC

- (void)viewDidLoad {

[super viewDidLoad];

self.title = @"错题本";

[self setRightItem];

self.magicView.navigationColor = OVERALLCOLOR;

self.magicView.sliderColor = BORDERCOLOR;

self.magicView.sliderWidth = 60;

self.magicView.layoutStyle = VTLayoutStyleCenter;

self.magicView.switchStyle = VTSwitchStyleDefault;

self.magicView.navigationHeight = 40.f;

self.magicView.itemScale = 1.1;

self.magicView.dataSource = self;

self.magicView.delegate = self;

self.magicView.needPreloading = NO;

[self.magicView reloadData];

}

- (void)setRightItem

{

NSArray \*arr = @[@"作业",@"自我测试",@"考试"];

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithTitle:arr[0] titleColor:[UIColor blackColor] andBarButtonItemBlock:^{

[FTPopOverMenu showFromSenderFrame:CGRectMake(self.view.frame.size.width - 40, 20, 40, 40)

withMenuArray:arr

imageArray:nil

doneBlock:^(NSInteger selectedIndex) {

self.navigationItem.rightBarButtonItem.title = arr[selectedIndex];

[DefaultCenter postNotificationName:kChangeErrorPaperTypeNotify object:@(selectedIndex)];

} dismissBlock:^{

NSLog(@"user canceled. do nothing.");

}];

}];

}

- (NSArray<NSString \*> \*)menuTitlesForMagicView:(VTMagicView \*)magicView {

return @[@"最新",@"分类"];

}

- (UIButton \*)magicView:(VTMagicView \*)magicView menuItemAtIndex:(NSUInteger)itemIndex {

static NSString \*itemIdentifier = @"ErrorPaperSonTVC";

UIButton \*menuItem = [magicView dequeueReusableItemWithIdentifier:itemIdentifier];

if (!menuItem) {

menuItem = [UIButton buttonWithType:UIButtonTypeCustom];

[menuItem setTitleColor:[UIColor colorWithHexString:@"#9a9a9a"] forState:UIControlStateNormal];

[menuItem setTitleColor:BORDERCOLOR forState:UIControlStateSelected];

menuItem.titleLabel.font = FONT(15);

}

return menuItem;

}

- (UIViewController \*)magicView:(VTMagicView \*)magicView viewControllerAtPage:(NSUInteger)pageIndex {

static NSString \*recomVCId = @"ErrorPaperSonTVC";

ErrorPaperSonTVC \*paperSonTVC = [magicView dequeueReusablePageWithIdentifier:recomVCId];

if (!paperSonTVC) {

paperSonTVC = [[ErrorPaperSonTVC alloc] initWithStyle:UITableViewStyleGrouped];

}

paperSonTVC.pageIndex = pageIndex;

if (pageIndex != 0) {

//显示 作业类型

[self setRightItem];

} else {

//隐藏 作业类型

self.navigationItem.rightBarButtonItem = nil;

//不是0 1 2就是最新

[DefaultCenter postNotificationName:kChangeErrorPaperTypeNotify object:@(-1)];

}

return paperSonTVC;

}

@end

//

// ErrorPaperSonTVC.m

// MaestroEdu

//

// Created by jesse on 2017/9/5.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "ErrorPaperSonTVC.h"

#import "ErrorPaperSonCell.h"

#import "ErrorPaperSonModel.h"

#import "PaperBrowseVC.h"

@interface ErrorPaperSonTVC ()

{

NSString \*\_filedName;

NSDictionary \*\_para;

}

@end

@implementation ErrorPaperSonTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.tableView.rowHeight = UITableViewAutomaticDimension;

self.tableView.estimatedRowHeight = 300;

//默认作业

\_filedName = @"0";

WEAKSELF

kTableViewFuckingInit(weakSelf\_SC.tableView, requestData);

[DefaultCenter addObserver:self selector:@selector(subjectRefreshA:) name:kChangeErrorPaperTypeNotify object:nil];

}

- (void)subjectRefreshA:(NSNotification \*)notify

{

\_filedName = [NSString stringWithFormat:@"%@",notify.object];

kTableViewHeaderAction(self.tableView);

[self requestData];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

kTableViewHeaderAction(self.tableView);

[self requestData];

}

- (void)requestData

{

\_para = @{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"5",@"filedName":\_filedName};

[JXHttpTool POST:@"getErrorsList" parameters:\_para success:^(id responseObject) {

kTableViewAfterDataAction(self.tableView, responseObject, [ErrorPaperSonModel parse:responseObject[@"outList"]])

} failure:nil];

}

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

{

return self.tableView.tableViewDatas.count;

}

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

{

ErrorPaperSonCell \*cell = [ErrorPaperSonCell cellWithNibTableView:tableView andReuseID:@"ErrorPaperSonCellSBID"];

[cell addDataWith:self.tableView.tableViewDatas[indexPath.row]];

return cell;

}

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

{

NSMutableDictionary \*muDict = \_para.mutableCopy;

[muDict setObject:@(indexPath.row + 1) forKey:@"incurrentPage"];

[muDict setObject:@(1) forKey:@"inshowCount"];

PaperBrowseVC \*paperBrowseVC = [[PaperBrowseVC alloc] init];

paperBrowseVC.muDict = muDict;

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

}

@end

//

// QuestionStuBankTVC.m

// MaestroEdu

//

// Created by 张明辉 on 2017/9/5.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "QuestionStuBankTVC.h"

#import "HomeTopView.h"

#import "FTPopOverMenu.h"

#import "QuestionBankCell.h"

#import "QuestionBankVC.h"

#import "QuestionStuBankModel.h"

#import "PaperBrowseVC.h"

#import "PaperListFinashTVC.h"

@interface QuestionStuBankTVC ()

{

NSMutableArray \*\_subjectArrs;

NSString \*\_proNumStr;

NSString \*\_paperNumStr;

}

@end

@implementation QuestionStuBankTVC

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

\_subjectArrs = @[].mutableCopy;

\_proNumStr = @"0";

\_paperNumStr = @"0";

self.view.backgroundColor = OVERALLCOLOR;

self.automaticallyAdjustsScrollViewInsets = NO;

WEAKSELF

kTableViewFuckingInit(weakSelf\_SC.tableView, requestData);

self.navigationItem.leftBarButtonItem = [UIBarButtonItem barButtonItemWithTitle:@"题库" titleColor:[UIColor whiteColor] andBarButtonItemBlock:^{

}];

//请求学科信息

[self requestSubjectData];

}

- (void)requestData

{

[JXHttpTool POST:@"testPaperTasklist" parameters:@{@"incurrentPage":self.tableView.CurrentPage?:@"1",@"inshowCount":@"20",@"inSubjectName":KACCOUNTTOOL.subjectImg?:EMPTY\_STRING} success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

SHOWERRORAFTERONESECOND(@"暂无数据");

kTableViewHeaderAction(self.tableView);

return ;

}

kTableViewAfterDataAction(self.tableView, responseObject, [QuestionStuBankModel parse:responseObject[@"outList"]]);

} failure:nil];

}

- (void)requestSubjectData

{

[JXHttpTool POST:@"subjectList" parameters:@{} success:^(id responseObject) {

NSMutableArray \*tempArr1 = [NSMutableArray arrayWithArray:responseObject[@"outList"]];

NSMutableArray \*tempArr = @[].mutableCopy;

for (NSDictionary \*dict in tempArr1) {

NSLog(@"%@",KACCOUNTTOOL.Faculty);

if ([dict[@"Faculty"] containsString:@"部"]) {

NSString \*temp = [dict[@"Faculty"] stringByReplacingOccurrencesOfString:@"部" withString:EMPTY\_STRING];

if ([temp isEqualToString:KACCOUNTTOOL.Faculty]) {

[tempArr addObject:dict];

}

}

}

\_subjectArrs = tempArr;

[self showSubjectList];

} failure:nil];

}

- (void)showSubjectList

{

NSMutableArray \*tempArr = @[].mutableCopy;

for (NSDictionary \*dict in \_subjectArrs) {

[tempArr addObject:dict[@"SubjectsName"]];

}

NSString \*showSubject;

//默认第一个

if (!KACCOUNTTOOL.subject) {

showSubject = tempArr[0];

KACCOUNTTOOL.subject = [NSString stringWithFormat:@"%@",\_subjectArrs[0][@"SubjectsId"]];

KACCOUNTTOOL.subjectImg = [NSString stringWithFormat:@"%@",\_subjectArrs[0][@"SubjectsName"]];

[KACCOUNTTOOL saveUserAccount];

[self postNotificationWith:\_subjectArrs[0]];

} else {

showSubject = KACCOUNTTOOL.subjectImg;

}

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:showSubject andBarButtonItemBlock:^{

[FTPopOverMenu showFromSenderFrame:CGRectMake(self.view.frame.size.width - 40, 20, 40, 40)

withMenuArray:tempArr

imageArray:nil

doneBlock:^(NSInteger selectedIndex) {

UIButton \*btn = (UIButton \*)self.navigationItem.rightBarButtonItem.customView;

[btn setBackgroundImage:IMG(tempArr[selectedIndex]) forState:UIControlStateNormal];

[btn setBackgroundImage:IMG(tempArr[selectedIndex]) forState:UIControlStateHighlighted];

NSLog(@"done block. do something. selectedIndex : %ld", (long)selectedIndex);

//切换科目 刷新

KACCOUNTTOOL.subject = [NSString stringWithFormat:@"%@",\_subjectArrs[selectedIndex][@"SubjectsId"]];

KACCOUNTTOOL.subjectImg = [NSString stringWithFormat:@"%@",\_subjectArrs[selectedIndex][@"SubjectsName"]];

[KACCOUNTTOOL saveUserAccount];

[self postNotificationWith:\_subjectArrs[selectedIndex]];

kTableViewHeaderAction(self.tableView);

[self requestData];

} dismissBlock:^{

NSLog(@"user canceled. do nothing.");

}];

}];

[DefaultCenter addObserver:self selector:@selector(subjectRefresh) name:kChangeSubjectNofity object:nil];

}

-(void)subjectRefresh

{

//刷新logo

UIButton \*btn = self.navigationItem.rightBarButtonItem.customView;

[btn setBackgroundImage:IMG(KACCOUNTTOOL.subjectImg) forState:UIControlStateNormal];

}

- (void)postNotificationWith:(NSDictionary \*)dict

{

[DefaultCenter postNotificationName:kChangeSubjectNofity object:nil];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBar.translucent = YES;

[self.navigationController.navigationBar setBackgroundImage:[[UIImage alloc] init] forBarMetrics:UIBarMetricsDefault];

self.navigationController.navigationBar.shadowImage = [[UIImage alloc] init];

if (\_subjectArrs

&& \_subjectArrs.count != 0) {

[self showSubjectList];

}

[self reqeustNumData];

[self requestData];

}

- (void)reqeustNumData

{

[JXHttpTool POST:@"allTestStatistics" parameters:@{} success:^(id responseObject) {

\_proNumStr = [NSString stringWithFormat:@"%@",responseObject[@"TestCount"]?:EMPTY\_STRING];

\_paperNumStr = [NSString stringWithFormat:@"%@",responseObject[@"TestAttrCount"]?:EMPTY\_STRING];

[self.tableView reloadData];

} failure:nil];

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBar.translucent = NO;

}

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

{

return 1 + self.tableView.tableViewDatas.count;

}

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

{

if (indexPath.row == 0) {

UITableViewCell \*cell = [tableView dequeueReusableCellWithIdentifier:@"financialCell"];

if (cell == nil) {

cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"financialCell"];

HomeTopView \*topView = [HomeTopView viewFromNIB];

topView.collectionImageView.image = IMG(@"icon\_questionbank\_white\_bottom");

topView.organizeImageView.image = IMG(@"icon\_questionbank\_white\_top");

[topView.organizeBtn setTitle:[NSString stringWithFormat:@"已做题:%@题",\_proNumStr?:@"0"] forState:UIControlStateNormal];

[topView.collectionBtn setTitle:[NSString stringWithFormat:@"已做卷:%@套",\_paperNumStr?:@"0"] forState:UIControlStateNormal];

topView.frame = cell.contentView.bounds;

//点击题 去 自我测试

topView.clickOrganizeBlock = ^{

QuestionBankVC \*questionBankVC = kStoryboardVC(@"QuestionBank", @"QuestionBankVCSBID");

questionBankVC.QuestionStuComeIn = YES;

questionBankVC.showTitle = @"选择范围";

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

};

//点击库

topView.clickCollectionBlock = ^{

// kPushStoryboardVC(@"Home", @"HomeCollectTVCSBID");

};

cell.selectionStyle = UITableViewCellSelectionStyleNone;

[cell.contentView addSubview:topView];

} else {

for (UIView \*view in cell.contentView.subviews) {

if ([view isKindOfClass:[HomeTopView class]]) {

HomeTopView \*topView = (HomeTopView \*)view;

[topView.organizeBtn setTitle:[NSString stringWithFormat:@"已做题:%@题",\_proNumStr?:@"0"] forState:UIControlStateNormal];

[topView.collectionBtn setTitle:[NSString stringWithFormat:@"已做卷:%@套",\_paperNumStr?:@"0"] forState:UIControlStateNormal];

}

}

}

return cell;

}

QuestionBankCell \*cell = [QuestionBankCell cellWithNibTableView:tableView andReuseID:@"QuestionBankCellSBID"];

[cell addDataWith:self.tableView.tableViewDatas[indexPath.row - 1]];

return cell;

}

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

{

if (indexPath.row == 0) {

return;

}

[super tableView:tableView didSelectRowAtIndexPath:indexPath];

//去做作业

QuestionStuBankModel \*model = self.tableView.tableViewDatas[indexPath.row - 1];

if ([model.Isstate isEqualToNumber:@2]

|| [model.Isstate isEqualToNumber:@1]) {

//完成 批改中

PaperListFinashTVC \*finashVC = [[PaperListFinashTVC alloc] init];

finashVC.inId = model.TestPaperId;

finashVC.inClassId = model.ClassId;

finashVC.inStudentId = model.StudentId;

if ([model.Isstate isEqualToNumber:@2]) {

finashVC.isCorrectting = NO;

finashVC.showTitle = @"练习回顾";

} else {

finashVC.isCorrectting = YES;

}

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

} else if ([model.Isstate isEqualToNumber:@0]) {

//领取任务

PaperBrowseVC \*paperBrowseVC = [[PaperBrowseVC alloc] init];

paperBrowseVC.HomeWorkStuComeIn = YES;

paperBrowseVC.inId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] TestPaperId]];

paperBrowseVC.StudentId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] StudentId]];

paperBrowseVC.ClassId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] ClassId]];

paperBrowseVC.SchoolId = [NSString stringWithFormat:@"%@",[self.tableView.tableViewDatas[indexPath.row - 1] SchoolId]];

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

}

}

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

{

if (indexPath.row == 0) {

return 200;

}

return 138;

}

@end

//

// LoginController1.m

// MaestroEdu

//

// Created by jesse on 2017/7/9.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "LoginController.h"

#import "JXTabBarController.h"

#import <UMSocialCore/UMSocialCore.h>

#import "MineModifyPwdTVC.h"

#import "RegistController.h"

@interface LoginController ()

@property (weak, nonatomic) IBOutlet UITextField \*phone;

@property (weak, nonatomic) IBOutlet JXPasswordTextField \*pwd;

@end

@implementation LoginController

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

self.phone.text = KACCOUNTTOOL.phone?:EMPTY\_STRING;//@"14725802580";//@"17665380688";//@"18068565639";// 18610715136;

self.pwd.text = EMPTY\_STRING;//@"111111";//@"ss123456";//;

self.phone.textColor = [UIColor whiteColor];

self.pwd.textColor = [UIColor whiteColor];

[self.phone setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

[self.pwd setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

}

- (IBAction)clickLogin:(UIButton \*)sender {

[JXHttpTool POST:@"Login" parameters:@{@"inNumber":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

KACCOUNTTOOL.outUser = [MineAccountOutUser parse:responseObject[@"outUser"]];

KACCOUNTTOOL.SchoolInfos = [MineAccountSchoolInfos parse:responseObject[@"SchoolInfos"]];

KACCOUNTTOOL.Token = responseObject[@"Token"];

[KACCOUNTTOOL afterLogin];

} failure:nil];

}

- (IBAction)clickForgetPwd:(UIButton \*)sender {

MineModifyPwdTVC \*pwd = kStoryboardVC(@"Mine", @"MineModifyPwdTVCSBID");

pwd.title = @"忘记密码";

[self.navigationController pushViewController:[JXBottomBtnController bottomBtnControllerWith:pwd bottomTitle:@"确定" andPara:@"1" gap:0] animated:YES];

}

- (IBAction)qqLogin:(UIButton \*)sender {

#warning 第三方登录完之后 完善信息，需要新增接口

[self getUserInfoForPlatform:UMSocialPlatformType\_QQ];

}

- (IBAction)wechatLogin:(UIButton \*)sender {

/\*

{

openid = "ocv0KwKYQWxP\_yEaJVUgG7J35FzQ";

city = "深圳";

country = "中国";

nickname = "Jesse";

privilege = [

];

language = "zh\_CN";

headimgurl = "http://wx.qlogo.cn/mmopen/Q3auHgzwzM4CynIamCusjAc0ASK1aTc0TDbgesr32ibx2w8ibsJa0WyECmVv1T1dmfTWl3CymIkq5685jiaR3lQ63iaJknaYJ5NkC75zo37VtibQ/0";

unionid = "oLsJhxHnikzGmttnemwhOlT0puzI";

sex = 1;

province = "广东"

}

\*/

[self getUserInfoForPlatform:UMSocialPlatformType\_WechatSession];

}

- (IBAction)sinaLogin:(UIButton \*)sender {

[self getUserInfoForPlatform:UMSocialPlatformType\_Sina];

}

- (void)getUserInfoForPlatform:(UMSocialPlatformType)platformType

{

[[UMSocialManager defaultManager] getUserInfoWithPlatform:platformType currentViewController:self completion:^(id result, NSError \*error) {

UMSocialUserInfoResponse \*resp = result;

// 第三方登录数据(为空表示平台未提供)

// 授权数据

NSLog(@" uid: %@", resp.uid);

NSLog(@" openid: %@", resp.openid);

NSLog(@" accessToken: %@", resp.accessToken);

NSLog(@" refreshToken: %@", resp.refreshToken);

NSLog(@" expiration: %@", resp.expiration);

// 用户数据

NSLog(@" name: %@", resp.name);

NSLog(@" iconurl: %@", resp.iconurl);

NSLog(@" gender: %@", resp.unionGender);

// 第三方平台SDK原始数据

NSLog(@" originalResponse: %@", resp.originalResponse);

if (!resp.openid) {

return ;

}

[JXHttpTool POST:@"thirdPartyLogin" parameters:@{@"inValue":resp.openid?:EMPTY\_STRING} success:^(id responseObject) {

//此账号未绑定 前去绑定

if ([responseObject[@"result"] isEqualToString:@"00"]) {

if ([responseObject[@"Isstate"] isEqualToNumber:@0]) {

//调用注册

//去完善信息

RegistController \*registVC = kStoryboardVC(@"LoginRegist", @"RegistControllerSBID");

registVC.isPerfectInfo = YES;

if (platformType == UMSocialPlatformType\_Sina) {

registVC.inSina = resp.openid;

} else if (platformType == UMSocialPlatformType\_QQ) {

registVC.inQQ = resp.openid;

} else if (platformType == UMSocialPlatformType\_WechatSession) {

registVC.inWeChat = resp.openid;

}

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

} else {

//直接登录

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

KACCOUNTTOOL.outUser = [MineAccountOutUser parse:responseObject[@"outUser"]];

KACCOUNTTOOL.SchoolInfos = [MineAccountSchoolInfos parse:responseObject[@"SchoolInfos"]];

KACCOUNTTOOL.Token = responseObject[@"Token"];

[KACCOUNTTOOL afterLogin];

}

}

} failure:nil];

}];

}

- (void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBarHidden = YES;

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBarHidden = NO;

}

@end

//

// RegistController2.m

// MaestroEdu

//

// Created by jesse on 2017/7/9.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "RegistController.h"

#import "MineAccountOutUser.h"

#import "DJCountDownTool.h"

@interface RegistController ()

@property (weak, nonatomic) IBOutlet UIButton \*loginNowBtn;

@property (weak, nonatomic) IBOutlet UITextField \*username;

@property (weak, nonatomic) IBOutlet JXPhoneTextField \*phone;

@property (weak, nonatomic) IBOutlet UITextField \*code;

@property (weak, nonatomic) IBOutlet JXPasswordTextField \*pwd;

@property (weak, nonatomic) IBOutlet UIButton \*backBtn;

@end

@implementation RegistController

- (void)viewDidLoad {

[super viewDidLoad];

// Do any additional setup after loading the view.

NSMutableAttributedString \*str = [[NSMutableAttributedString alloc] initWithString:@"立即登录"];

NSRange strRange = {0,[str length]};

[str addAttributes:@{NSUnderlineStyleAttributeName:[NSNumber numberWithInteger:NSUnderlineStyleSingle],NSForegroundColorAttributeName:WHITECOLOR} range:strRange];

[self.loginNowBtn setAttributedTitle:str forState:UIControlStateNormal];

if (self.isPerfectInfo) {

self.title = @"完善信息";

self.backBtn.hidden = NO;

}

self.phone.textColor = [UIColor whiteColor];

self.pwd.textColor = [UIColor whiteColor];

[self.phone setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

[self.pwd setValue:[UIColor whiteColor] forKeyPath:@"\_placeholderLabel.textColor"];

}

- (IBAction)clickCode:(UIButton \*)sender {

[DJCountDownTool sendVertifyWithBtn:sender phone:self.phone.text?:EMPTY\_STRING andType:@"0"];

}

- (IBAction)clickBackBtn:(UIButton \*)sender {

[self.navigationController popViewControllerAnimated:YES];

}

- (IBAction)clickRegist:(UIButton \*)sender {

kJudgePhone(self.phone.text);

kJudgePassword(self.pwd.text);

kJudgeNullWith(self.username.text, @"请输入用户名");

kJudgeNullWith(self.code.text, @"请输入验证码");

if (self.isPerfectInfo) {

//完善信息

NSDictionary \*dict;

if (self.inSina) {

dict = @{@"inUserName":self.username.text?:EMPTY\_STRING,@"inTel":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING,@"inSMScode":self.code.text?:EMPTY\_STRING,@"inSina":self.inSina?:EMPTY\_STRING};

} else if (self.inWeChat) {

dict = @{@"inUserName":self.username.text?:EMPTY\_STRING,@"inTel":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING,@"inSMScode":self.code.text?:EMPTY\_STRING,@"inWeChat":self.inWeChat?:EMPTY\_STRING};

} else if (self.inQQ) {

dict = @{@"inUserName":self.username.text?:EMPTY\_STRING,@"inTel":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING,@"inSMScode":self.code.text?:EMPTY\_STRING,@"inQQ":self.inQQ?:EMPTY\_STRING};

}

[JXHttpTool POST:@"teacherRegister" parameters:dict success:^(id response) {

[self login];

} failure:nil tableView:nil result:nil needToken:NO];

} else {

[JXHttpTool POST:sender urlString:@"studentRegister" parameters:@{@"inUserName":self.username.text?:EMPTY\_STRING,@"inTel":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING,@"inSMScode":self.code.text?:EMPTY\_STRING} success:^(id responseObject) {

NSLog(@"%@",responseObject);

SHOWSUCCESSAFTERONESECOND(@"注册成功");

// [self login];

[self.navigationController popViewControllerAnimated:YES];

} failure:nil];

}

}

- (void)login

{

[JXHttpTool POST:@"Login" parameters:@{@"inNumber":self.phone.text?:EMPTY\_STRING,@"inPassword":self.pwd.text?:EMPTY\_STRING} success:^(id responseObject) {

KACCOUNTTOOL.outUser = [MineAccountOutUser parse:responseObject[@"outUser"]];

KACCOUNTTOOL.SchoolInfos = [MineAccountSchoolInfos parse:responseObject[@"SchoolInfos"]];

[KACCOUNTTOOL afterLogin];

} failure:nil];

}

- (IBAction)clickLogin:(UIButton \*)sender {

[self.navigationController popViewControllerAnimated:YES];

}

- (void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

self.navigationController.navigationBarHidden = YES;

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

self.navigationController.navigationBarHidden = NO;

}

/\*

#pragma mark - Navigation

// In a storyboard-based application, you will often want to do a little preparation before navigation

- (void)prepareForSegue:(UIStoryboardSegue \*)segue sender:(id)sender {

// Get the new view controller using [segue destinationViewController].

// Pass the selected object to the new view controller.

}

\*/

@end

//

// NSDictionary+LogCategory.m

// TestMac

//

// Created by yuelixing on 16/9/5.

// Copyright © 2016年 Tutu. All rights reserved.

//

#import "LogCategory.h"

#import <objc/runtime.h>

static NSString \* const kLxArrayBegin = @"[";

static NSString \* const kLxArrayEnd = @"]";

static NSString \* const kLxDictionaryBegin = @"{";

static NSString \* const kLxDictionaryEnd = @"}";

static NSString \* const kLxSetBegin = @"{(";

static NSString \* const kLxSetEnd = @")}";

@implementation NSArray (Log)

#ifdef UseLogChinese

- (NSString \*)debugDescription {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)description {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)descriptionWithLocale:(id)locale {

NSMutableString \* string = [NSMutableString string];

[string appendFormat:@"%@\n", kLxArrayBegin];

NSUInteger count = self.count;

[self enumerateObjectsUsingBlock:^(id \_Nonnull obj, NSUInteger idx, BOOL \* \_Nonnull stop) {

NSString \* temp = nil;

if ([obj respondsToSelector:@selector(descriptionWithLocale:)]) {

temp = [obj performSelector:@selector(descriptionWithLocale:) withObject:locale];

temp = [temp stringByReplacingOccurrencesOfString:@"\n" withString:@"\n\t"];

} else {

temp = [obj performSelector:@selector(description) withObject:nil];

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

temp = [NSString stringWithFormat:@"\"%@\"", temp];

}

}

[string appendFormat:@"\t%@", temp];

if (idx+1 != count) {

[string appendString:@","];

}

[string appendString:@"\n"];

}];

[string appendString:kLxArrayEnd];

return string;

}

#endif

@end

@implementation NSDictionary (Log)

#ifdef UseLogChinese

- (NSString \*)debugDescription {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)description {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)descriptionWithLocale:(id)locale {

NSMutableString \* string = [NSMutableString string];

[string appendFormat:@"%@\n", kLxDictionaryBegin];

NSUInteger count = self.allKeys.count;

for (id key in self.allKeys) {

NSInteger index = [self.allKeys indexOfObject:key];

id value = [self objectForKey:key];

NSString \* temp = nil;

if ([value respondsToSelector:@selector(descriptionWithLocale:)]) {

temp = [value performSelector:@selector(descriptionWithLocale:) withObject:locale];

temp = [temp stringByReplacingOccurrencesOfString:@"\n" withString:@"\n\t"];

} else {

temp = [value performSelector:@selector(description) withObject:nil];

if ([value isKindOfClass:[NSString class]]) {

temp = [NSString stringWithFormat:@"\"%@\"", temp];

}

}

[string appendFormat:@"\t%@ = %@", key, temp];

if (index+1 != count) {

[string appendString:@";"];

}

[string appendString:@"\n"];

}

[string appendString:kLxDictionaryEnd];

return string;

}

#endif

@end

@implementation NSSet (Log)

#ifdef UseLogChinese

- (NSString \*)debugDescription {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)description {

return [NSString stringWithFormat:@"<%@ %p> %@", NSStringFromClass([self class]), self, [self descriptionWithLocale:nil]];

}

- (NSString \*)descriptionWithLocale:(id)locale {

NSMutableString \* string = [NSMutableString string];

[string appendFormat:@"%@\n", kLxSetBegin];

NSUInteger count = self.count;

\_\_block NSInteger idx = 0;

[self enumerateObjectsUsingBlock:^(id \_Nonnull obj, BOOL \* \_Nonnull stop) {

NSString \* temp = nil;

if ([obj respondsToSelector:@selector(descriptionWithLocale:)]) {

temp = [obj performSelector:@selector(descriptionWithLocale:) withObject:locale];

temp = [temp stringByReplacingOccurrencesOfString:@"\n" withString:@"\n\t"];

} else {

temp = [obj performSelector:@selector(description) withObject:nil];

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

temp = [NSString stringWithFormat:@"\"%@\"", temp];

}

}

[string appendFormat:@"\t%@", temp];

if (idx+1 != count) {

[string appendString:@","];

}

[string appendString:@"\n"];

idx += 1;

}];

[string appendString:kLxSetEnd];

return string;

}

#endif

@end

//

// UIBarButtonItem+Category.m

// ThePatient

//

// Created by 张明辉 on 16/8/23.

// Copyright © 2016年 jesse. All rights reserved.

//

#import "UIBarButtonItem+Category.h"

#import <objc/runtime.h>

#import "UIButton+Awesome.h"

@interface UIBarButtonItem ()

@property (nonatomic) BarButtonItemBlock barButtonItemBlock;

@end

@implementation UIBarButtonItem (Category)

+(instancetype)barButtonItemWithImage:(NSString \*)image andBarButtonItemBlock:(BarButtonItemBlock)barButtonItemBlock

{

UIButton \*btn = [UIButton buttonWithImageName:image];

UIBarButtonItem \*item = [[UIBarButtonItem alloc] initWithCustomView:btn];

item.barButtonItemBlock = barButtonItemBlock;

[btn addTarget:item action:@selector(clickItem) forControlEvents:UIControlEventTouchUpInside];

return item;

}

+(instancetype)barButtonItemWithTitle:(NSString \*)title andBarButtonItemBlock:(BarButtonItemBlock)barButtonItemBlock

{

//默认是白色文字

return [self barButtonItemWithTitle:title titleColor:[UIColor blackColor] andBarButtonItemBlock:barButtonItemBlock];

}

+(instancetype)barButtonItemWithTitle:(NSString \*)title titleColor:(UIColor \*)titleColor andBarButtonItemBlock:(BarButtonItemBlock)barButtonItemBlock

{

UIBarButtonItem \*item = [[UIBarButtonItem alloc] initWithTitle:title style:UIBarButtonItemStylePlain target:nil action:nil];

[item setTarget:item];

[item setAction:@selector(clickItem)];

item.barButtonItemBlock = barButtonItemBlock;

item.tintColor = titleColor;

return item;

}

- (void)clickItem

{

if (self.barButtonItemBlock) {

self.barButtonItemBlock();

}

}

static void \*key = &key;

-(void)setBarButtonItemBlock:(BarButtonItemBlock)barButtonItemBlock

{

objc\_setAssociatedObject(self, & key, barButtonItemBlock, OBJC\_ASSOCIATION\_COPY);

}

-(BarButtonItemBlock)barButtonItemBlock

{

return objc\_getAssociatedObject(self, &key);

}

@end

//

// DJCountDownTool.m

// DJMedical

//

// Created by 张明辉 on 16/12/18.

// Copyright © 2016年 jesse. All rights reserved.

//

#import "DJCountDownTool.h"

@interface DJCountDownTool ()

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

@property (nonatomic) int countDownTime;

@end

@implementation DJCountDownTool

+(void)getVertifyCountDownWithBtn:(UIButton \*)getVertifyBtn

{

//执行倒计时

getVertifyBtn.enabled = NO;

DJCountDownTool \*getVertify = [[DJCountDownTool alloc] init];

getVertify.getVertifyBtn = getVertifyBtn;

getVertify.countDownTime = 120;

getVertify.timer = [NSTimer scheduledTimerWithTimeInterval:1.f target:getVertify selector:@selector(countDown) userInfo:nil repeats:YES];

}

- (void)countDown

{

[self.getVertifyBtn setTitle:[NSString stringWithFormat:@"(%dS)",self.countDownTime] forState:UIControlStateNormal];

self.countDownTime --;

if (self.countDownTime == -1) {

[self.timer invalidate];

self.timer = nil;

[self.getVertifyBtn setTitle:@"重新获取" forState:UIControlStateNormal];

self.getVertifyBtn.enabled = YES;

// self.getVertifyBtn.backgroundColor = KWHITECOLOR;

}

}

+(instancetype)countDownWithBtn:(UIButton \*)btn andCountDownDIYEvents:(CountDownDIYEvents)countDownDIYEvents

{

DJCountDownTool \*getVertify = [[DJCountDownTool alloc] init];

getVertify.getVertifyBtn = btn;

getVertify.countDownTime = 0;

if (countDownDIYEvents) {

getVertify.diyEvent = countDownDIYEvents;

}

getVertify.timer = [NSTimer scheduledTimerWithTimeInterval:1.f target:getVertify selector:@selector(countDownDIY) userInfo:nil repeats:YES];

return getVertify;

}

- (void)countDownDIY

{

[self.getVertifyBtn setTitle:[NSString stringWithFormat:@"%ds",self.countDownTime] forState:UIControlStateNormal];

self.countDownTime --;

if (self.countDownTime == -1) {

[self.timer invalidate];

self.timer = nil;

if (self.diyEvent) {

self.diyEvent();

}

self.getVertifyBtn.enabled = YES;

}

}

/\*\*

\* 获取验证码

\*/

+ (void)sendVertifyWithBtn:(UIButton \*)sender phone:(NSString \*)phone andType:(NSString \*)type

{

kJudgePhone(phone);

//0:注册 1:忘记密码 2:修改密码 3:更换手机。

// 当inType为3时，inTel为原手机号

[JXHttpTool POST:@"regcode" parameters:@{@"inTel":phone?:EMPTY\_STRING,@"inType":type?:EMPTY\_STRING} success:^(id response) {

SHOWSUCCESSAFTERONESECOND(@"发送成功!")

[DJCountDownTool getVertifyCountDownWithBtn:sender];

} failure:nil tableView:nil result:nil needToken:NO];

}

@end

//

// DJUtils.m

// DJMedical

//

// Created by 张明辉 on 16/12/21.

// Copyright © 2016年 jesse. All rights reserved.

//

#import "DJUtils.h"

#import "LoginController.h"

@implementation DJUtils

#pragma 没登录 则模态登陆

+(BOOL)willShowLoginWithController:(UIViewController \*)viewController

{

return [DJUtils willShowLoginWithController:viewController afterLogin:nil];

}

+(BOOL)willShowLoginWithController:(UIViewController \*)viewController afterLogin:(void (^)())afterLogin

{

// //如果没登录 模态登录界面

// if (![KACCOUNTTOOL isLogin]) {

// LoginController \*loginVC = [UIStoryboard initialViewControllerWithSbName:@"LoginAndRegist" andSBID:@"LoginVCSBID"];

// JXBaseNavigationController \*nav = [[JXBaseNavigationController alloc] initWithRootViewController:loginVC];

//

// [viewController presentViewController:nav animated:YES completion:nil];

// return YES;

// }

return NO;

}

#pragma 正则匹配手机号

+ (BOOL)checkTelNumber:(NSString \*) telNumber

{

NSString \*pattern =@"^1[3|4|5|7|8][0-9]\\d{8}";

NSPredicate \*pred = [NSPredicate predicateWithFormat:@"SELF MATCHES %@", pattern];

BOOL isMatch = [pred evaluateWithObject:telNumber];

return isMatch;

}

#pragma 正则匹配用户密码6-16位数字或字母组合

+ (BOOL)checkPassword:(NSString \*) password

{

//必须包含数字和字母

NSString \*pattern = @"^(?![0-9]\*$)[a-zA-Z0-9]{6,16}$";

//纯数字或纯字母都可以

// NSString \*pattern = @"[0-9a-zA-Z]{6,16}";

NSPredicate \*pred = [NSPredicate predicateWithFormat:@"SELF MATCHES %@", pattern];

BOOL isMatch = [pred evaluateWithObject:password];

return isMatch;

}

//#pragma 匹配用户密码6-18位随便字符

//+ (BOOL)checkPassword:(NSString \*) password

//{

// BOOL isOK;

// if (password.length >= 6 && password.length <= 18) {

// isOK = YES;

// } else {

// isOK = NO;

// }

// return isOK;

//}

///合法身份证

+ (BOOL)isValidPersonIDCardNumber:(NSString \*)number

{

return [DJUtils isHxIdentityCardWithIdCard:number];

}

// 匹配身份证号码

+(BOOL)isHxIdentityCardWithIdCard:(NSString \*)idCard

{

// 判断位数

if ([idCard length] != 15 && [idCard length] != 18)

{

return NO;

}

NSString \*carid = idCard;

long lSumQT =0;

// 加权因子

int R[] ={7, 9, 10, 5, 8, 4, 2, 1, 6, 3, 7, 9, 10, 5, 8, 4, 2 };

// 校验码

unsigned char sChecker[11]={'1','0','X', '9', '8', '7', '6', '5', '4', '3', '2'};

// 将15位身份证号转换成18位

NSMutableString \*mString = [NSMutableString stringWithString:idCard];

if ([idCard length] == 15)

{

[mString insertString:@"19" atIndex:6];

long p = 0;

const char \*pid = [mString UTF8String];

for (int i=0; i<=16; i++)

{

p += (pid[i]-48) \* R[i];

}

int o = p%11;

NSString \*string\_content = [NSString stringWithFormat:@"%c",sChecker[o]];

[mString insertString:string\_content atIndex:[mString length]];

carid = mString;

}

// 判断地区码

NSString \* sProvince = [carid substringToIndex:2];

if (![DJUtils areaCode:sProvince])

{

return NO;

}

// 判断年月日是否有效

// 年份

int strYear = [[carid substringWithRange:NSMakeRange(6,4)] intValue];

// 月份

int strMonth = [[carid substringWithRange:NSMakeRange(10,2)] intValue];

// 日

int strDay = [[carid substringWithRange:NSMakeRange(12,2)] intValue];

NSTimeZone \*localZone = [NSTimeZone localTimeZone];

NSDateFormatter \*dateFormatter = [[NSDateFormatter alloc] init];

[dateFormatter setDateStyle:NSDateFormatterMediumStyle];

[dateFormatter setTimeStyle:NSDateFormatterNoStyle];

[dateFormatter setTimeZone:localZone];

[dateFormatter setDateFormat:@"yyyy-MM-dd HH:mm:ss"];

NSDate \*date=[dateFormatter dateFromString:[NSString stringWithFormat:@"%d-%d-%d 12:01:01",strYear,strMonth,strDay]];

if (date == nil)

{

return NO;

}

const char \*PaperId = [carid UTF8String];

// 检验长度

if( 18 != strlen(PaperId)) return -1;

// 校验数字

for (int i=0; i<18; i++)

{

if ( !isdigit(PaperId[i]) && !(('X' == PaperId[i] || 'x' == PaperId[i]) && 17 == i) )

{

return NO;

}

}

// 验证最末的校验码

for (int i=0; i<=16; i++)

{

lSumQT += (PaperId[i]-48) \* R[i];

}

if (sChecker[lSumQT%11] != PaperId[17] )

{

return NO;

}

return YES;

}

+(BOOL)areaCode:(NSString \*)code

{

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

[dic setObject:@"北京" forKey:@"11"];

[dic setObject:@"天津" forKey:@"12"];

[dic setObject:@"河北" forKey:@"13"];

[dic setObject:@"山西" forKey:@"14"];

[dic setObject:@"内蒙古" forKey:@"15"];

[dic setObject:@"辽宁" forKey:@"21"];

[dic setObject:@"吉林" forKey:@"22"];

[dic setObject:@"黑龙江" forKey:@"23"];

[dic setObject:@"上海" forKey:@"31"];

[dic setObject:@"江苏" forKey:@"32"];

[dic setObject:@"浙江" forKey:@"33"];

[dic setObject:@"安徽" forKey:@"34"];

[dic setObject:@"福建" forKey:@"35"];

[dic setObject:@"江西" forKey:@"36"];

[dic setObject:@"山东" forKey:@"37"];

[dic setObject:@"河南" forKey:@"41"];

[dic setObject:@"湖北" forKey:@"42"];

[dic setObject:@"湖南" forKey:@"43"];

[dic setObject:@"广东" forKey:@"44"];

[dic setObject:@"广西" forKey:@"45"];

[dic setObject:@"海南" forKey:@"46"];

[dic setObject:@"重庆" forKey:@"50"];

[dic setObject:@"四川" forKey:@"51"];

[dic setObject:@"贵州" forKey:@"52"];

[dic setObject:@"云南" forKey:@"53"];

[dic setObject:@"西藏" forKey:@"54"];

[dic setObject:@"陕西" forKey:@"61"];

[dic setObject:@"甘肃" forKey:@"62"];

[dic setObject:@"青海" forKey:@"63"];

[dic setObject:@"宁夏" forKey:@"64"];

[dic setObject:@"新疆" forKey:@"65"];

[dic setObject:@"台湾" forKey:@"71"];

[dic setObject:@"香港" forKey:@"81"];

[dic setObject:@"澳门" forKey:@"82"];

[dic setObject:@"国外" forKey:@"91"];

if ([dic objectForKey:code] == nil) {

return NO;

}

return YES;

}

+ (BOOL)isNumber

{

NSPredicate \*pred = [NSPredicate predicateWithFormat:@"SELF MATCHES %@", @"^[0-9]+$"];

BOOL isnNum = [pred evaluateWithObject:self];

return isnNum;

}

//银行卡校验规则(Luhn算法)

+ (BOOL)isValidBankCardNumber:(NSString \*)number

{

NSInteger len = [number length];

NSInteger sumNumOdd = 0;

NSInteger sumNumEven = 0;

BOOL isOdd = YES;

for (NSInteger i = len - 1; i >= 0; i--) {

NSInteger num = [number substringWithRange:NSMakeRange(i, 1)].integerValue;

if (isOdd) {//奇数位

sumNumOdd += num;

}else{//偶数位

num = num \* 2;

if (num > 9) {

num = num - 9;

}

sumNumEven += num;

}

isOdd = !isOdd;

}

return ((sumNumOdd + sumNumEven) % 10 == 0);

}

+ (BOOL)validateEmailAddress:(NSString \*)address

{

NSString \*emailCheck = @"[A-Z0-9a-z.\_%+-]+@[A-Za-z0-9.-]+\\.[A-Za-z]{2,8}";

NSPredicate \*emailMatch = [NSPredicate predicateWithFormat:@"SELF MATCHES%@",emailCheck];

if (![emailMatch evaluateWithObject:address]) {

return NO;

}

return YES;

}

+ (BOOL)isAlphanumericWith:(NSString \*)str

{

NSPredicate \*pred = [NSPredicate predicateWithFormat:@"SELF MATCHES %@", @"^[a-zA-Z0-9]+$"];

return [pred evaluateWithObject:str];

}

/\*\*

\*颜色值转换成图片

\*/

+ (UIImage\*) createImageWithColor: (UIColor\*) color

{

CGRect rect=CGRectMake(0.0f, 0.0f, 1.0f, 1.0f);

UIGraphicsBeginImageContext(rect.size);

CGContextRef context = UIGraphicsGetCurrentContext();

CGContextSetFillColorWithColor(context, [color CGColor]);

CGContextFillRect(context, rect);

UIImage\*theImage = UIGraphicsGetImageFromCurrentImageContext();

UIGraphicsEndImageContext();

return theImage;

}

+(NSString \*)getSexNumWithSex:(NSString \*)sex

{

//性别 0：男 1：女

NSString \*temp = sex?:EMPTY\_STRING;

if ([sex isEqualToString:@"男"]) {

temp = @"0";

} else if ([sex isEqualToString:@"女"]) {

temp = @"1";

}

return temp;

}

+(NSString \*)getSexWithSexNum:(NSString \*)sexNum

{

//性别 0：男 1：女

NSString \*temp = sexNum?:EMPTY\_STRING;

if ([sexNum isEqualToString:@"0"]) {

temp = @"男";

} else if ([sexNum isEqualToString:@"1"]) {

temp = @"女";

}

return temp;

}

//传身份证返回生日字符串

+ (NSString \*)birthdayStrFromIdentityCard:(NSString \*)numberStr{

NSMutableString \*result = [NSMutableString stringWithCapacity:0];

NSString \*year = nil;

NSString \*month = nil;

BOOL isAllNumber = YES;

NSString \*day = nil;

if([numberStr length]<14)

return result;

//\*\*截取前14位

NSString \*fontNumer = [numberStr substringWithRange:NSMakeRange(0, 13)];

//\*\*检测前14位否全都是数字;

const char \*str = [fontNumer UTF8String];

const char \*p = str;

while (\*p!='\0') {

if(!(\*p>='0'&&\*p<='9'))

isAllNumber = NO;

p++;

}

if(!isAllNumber)

return result;

year = [numberStr substringWithRange:NSMakeRange(6, 4)];

month = [numberStr substringWithRange:NSMakeRange(10, 2)];

day = [numberStr substringWithRange:NSMakeRange(12,2)];

[result appendString:year];

[result appendString:@""];

[result appendString:month];

[result appendString:@""];

[result appendString:day];

return result;

}

//传入 秒 得到 xx分钟xx秒

+(NSString \*)getMMSSFromSS:(NSString \*)totalTime{

NSInteger seconds = [totalTime integerValue];

//format of minute

NSString \*str\_minute = [NSString stringWithFormat:@"%02ld",seconds/60];

//format of second

NSString \*str\_second = [NSString stringWithFormat:@"%02ld",seconds%60];

//format of time

NSString \*format\_time = [NSString stringWithFormat:@"%@:%@",str\_minute,str\_second];

return format\_time;

}

@end

//

// JXAccountTool.m

// CourierApp

//

// Created by 张明辉 on 17/2/24.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "JXAccountTool.h"

#import "MJExtension.h"

#import "AppDelegate.h"

#import "MineBandScroolVC.h"

//#import "DJLoginVC.h"

//#import "DJTabBarController.h"

//#import "JPUSHService.h"

#define kUserInfo @"userInfo"

#import "NSString+Awesome.h"

#define kUserFilePath [NSString filePathForDocuments:kUserInfo]

@implementation JXAccountTool

MJCodingImplementation

+(instancetype)sharedTool

{

static JXAccountTool \*tool;

static dispatch\_once\_t onceToken;

dispatch\_once(&onceToken, ^{

tool = [NSKeyedUnarchiver unarchiveObjectWithFile:kUserFilePath];

if (!tool) {

tool = [[JXAccountTool alloc] init];

tool.Faculty = @"高中";

}

});

return tool;

}

- (void)logOut

{

self.isLogin = NO;

self.phone = KACCOUNTTOOLOUTUSER.Phone;

self.outUser = [[MineAccountOutUser alloc] init];

[self.SchoolInfos removeAllObjects];

self.Faculty = @"高中";

[self saveUserAccount];

if ([ROOTTABBARVC isKindOfClass:[UITabBarController class]]) {

[ROOTTABBARVC setSelectedIndex:0];

}

UIViewController \*loginVC = kStoryboardVC(@"LoginRegist", @"LoginControllerSBID");

JXNavigationController \*nav = [[JXNavigationController alloc] initWithRootViewController:loginVC];

APPD.keyWindow.rootViewController = nav;

}

#pragma mark 请求道用户详细数据之后 保存

- (void)saveUserAccount {

[NSKeyedArchiver archiveRootObject:self toFile:kUserFilePath];

}

#pragma mark 退出之后 清除数据

- (void)afterLogOut {

[self logOut];

}

- (void)afterLogin {

[JXAccountTool refreshUserRoleSuccess:^(id responseObject) {

//先判断是否绑定过学校

if (![UserDefault objectForKey:KSkipBandSchool]) {

if ([KACCOUNTTOOLOUTUSER.IdCard isEqualToString:EMPTY\_STRING]) {

//去绑定学校

MineBandScroolVC \*bandVC = kStoryboardVC(@"Mine", @"MineBandScroolVCSBID");

bandVC.isLoginComeIn = YES;

[(UINavigationController \*)[self getCurrentVC] pushViewController:bandVC animated:YES];

return;

}

}

AppDelegate \* appDelegate = (AppDelegate \*)[UIApplication sharedApplication].delegate;

APPD.keyWindow.rootViewController = (UIViewController \*)appDelegate.tabbarVC;

[ROOTTABBARVC setSelectedIndex:0];

self.isLogin = YES;

[self saveUserAccount];

[DefaultCenter postNotificationName:AfterLoginRefresh object:nil];

}];

}

+ (void)refreshUserInfoSuccess:(void (^)( id responseObject))success

{

[JXHttpTool POST:@"userList" parameters:@{@"Token":KACCOUNTTOOLTOKEN,@"inNumber":KACCOUNTTOOLOUTUSER.PlatformNumber} success:^(id responseObject) {

NSLog(@"%@",responseObject);

[JXAccountTool sharedTool].outUser = [MineAccountOutUser parse:responseObject[@"outUser"]];

[JXAccountTool sharedTool].outSchoolInfo = [MineAccountSchoolInfos parse:responseObject[@"outSchoolInfo"]];

// [KACCOUNTTOOL saveUserAccount];

if (success) {

success(responseObject);

}

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

+ (void)refreshUserRoleSuccess:(void (^)( id responseObject))success

{

[JXHttpTool POST:@"userRoleList" parameters:@{@"Token":KACCOUNTTOOLTOKEN} success:^(id responseObject) {

[JXAccountTool sharedTool].roleArr = [NSMutableArray arrayWithArray:responseObject[@"outList"]];

[[JXAccountTool sharedTool] saveUserAccount];

if (success) {

success(responseObject);

}

} failure:^(id error) {

NSLog(@"%@",error);

}];

}

- (BOOL)haveQuestionManagerRole

{

BOOL Question = NO;

for (NSDictionary \*dict in KACCOUNTTOOL.roleArr) {

if ([dict[@"MenuName"] isEqualToString:@"题库管理"]) {

if ([dict[@"BtnOpt"] isEqualToNumber:@1]) {

Question = YES;

} else {

Question = NO;

}

}

}

return Question;

}

- (BOOL)haveRecordManagerRole

{

BOOL record = NO;

for (NSDictionary \*dict in KACCOUNTTOOL.roleArr) {

if ([dict[@"MenuName"] isEqualToString:@"记录管理"]) {

if ([dict[@"BtnOpt"] isEqualToNumber:@1]) {

record = YES;

} else {

record = NO;

}

}

}

return record;

}

@end

//

// PaperBrowseVC.m

// MaestroEdu

//

// Created by jesse on 2017/7/11.

// Copyright © 2017年 jesse. All rights reserved.

//

#import "PaperBrowseVC.h"

#import "PaperObjectView.h"

#import "FTPopOverMenu.h"

#import "PaperBrowseModel.h"

#import "PaperBrowseProblemModel.h"

#import "PaperBrowseQuestionsModel.h"

#import "PaperBrowseOptionModel.h"

#import "SunOrderArray.h"

#import "PaperBrowseCollectListView.h"

#import "UISlipUpTool.h"

#import "HomeCollectCreateAndEdit.h"

#import "ErrorComeDetailRightItem.h"

#import "TestMySelfLeftItem.h"

#import "TestMySelfRightItem.h"

#import "PaperInsertTypeView.h"

#import "PaperListVC.h"

#import "PaperCorrectMistakeTVC.h"

#import "SnShareTool.h"

@interface PaperBrowseVC ()<UIScrollViewDelegate>

{

NSInteger \_currentPage;

NSMutableArray \*\_allDatasArr;

PaperBrowseModel \*\_currentModel;

int \_allTime;

NSInteger \_homeWordCurrentPage;

NSMutableDictionary \*\_allSuccessPro;//已答题的数组

}

//最大的scrollview

@property (nonatomic,strong) UIScrollView \*scrollView;

@property (nonatomic,strong) PaperBrowseModel \*model;

@property (nonatomic,strong) NSTimer \*timer;

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

@end

@implementation PaperBrowseVC

- (void)viewDidLoad {

[super viewDidLoad];

self.title = @"试题浏览";

\_currentPage = 0;

\_allDatasArr = @[].mutableCopy;

\_allTime = 0;

\_homeWordCurrentPage = 1;

\_allSuccessPro = @{}.mutableCopy;

if (self.paperBrowerListComeIn) {

self.title = @"练习详情";

}

if (self.HomeWorkStuComeIn || self.QuestionStuComeIn) {

self.title = @"做作业";

if (self.QuestionStuComeIn) {

self.title = @"自我测试";

}

self.answerDict = @{@"Token":KACCOUNTTOOLTOKEN,@"inImgAnswer":@[].mutableCopy,@"inImgAnswerObj":@[].mutableCopy,@"inStudentId":self.StudentId?:EMPTY\_STRING,@"inClassId":self.ClassId?:EMPTY\_STRING,@"inSchoolId":self.SchoolId?:EMPTY\_STRING,@"inAttributeId":self.inId?:EMPTY\_STRING,@"inAnswerObj":@[].mutableCopy}.mutableCopy;

}

if (self.inFavId) {

//收藏列表进入的话

self.title = @"收藏夹";

TestMySelfLeftItem \*leftItem = [TestMySelfLeftItem viewFromNIB];

[leftItem.rightBtn setImage:IMG(@"general\_icon\_operat") forState:UIControlStateNormal];

leftItem.clickLeftBlock = ^{

[self.navigationController popViewControllerAnimated:YES];

};

leftItem.clickRightBlock = ^{

PaperListVC \*listVC = [[PaperListVC alloc] init];

listVC.refreshPageBlock = ^(int index) {

\_currentPage = index;

[self.scrollView scrollRectToVisible:CGRectMake(\_currentPage \* SCREEN\_WIDTH, 0, SCREEN\_WIDTH, SCREEN\_HEIGHT) animated:NO];

\_currentModel = \_allDatasArr[\_currentPage];

};

listVC.isBrowerPaperListCollectComeIn = YES;

listVC.inFavId = self.inFavId?:EMPTY\_STRING;

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

};

self.navigationItem.leftBarButtonItem = [[UIBarButtonItem alloc] initWithCustomView:leftItem];

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"general\_icon\_more" andBarButtonItemBlock:^{

[self showList];

}];

} else if (self.muDict) {

//错题本的话

ErrorComeDetailRightItem \*rightItem = [ErrorComeDetailRightItem viewFromNIB];

rightItem.clickLeftBlock = ^(UIButton \*sender){

//点击攻克

sender.selected = !sender.selected;

if (sender.selected) {

//攻克

[JXHttpTool POST:@"addConquer" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inTestId":\_currentModel.TestId?:EMPTY\_STRING,@"inConquer":@"1"} success:^(id responseObject) {

for (UIView \*view in self.scrollView.subviews) {

if ([view isKindOfClass:[PaperObjectView class]]) {

PaperObjectView \*objView = (PaperObjectView \*)view;

objView.showConquer.hidden = NO;

}

}

// \_currentPage

} failure:nil];

} else {

//取消攻克

[JXHttpTool POST:@"addConquer" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inTestId":\_currentModel.TestId?:EMPTY\_STRING,@"inConquer":@"0"} success:^(id responseObject) {

for (UIView \*view in self.scrollView.subviews) {

if ([view isKindOfClass:[PaperObjectView class]]) {

PaperObjectView \*objView = (PaperObjectView \*)view;

objView.showConquer.hidden = YES;

}

}

} failure:nil];

}

};

rightItem.clickCenterBlock = ^{

//点击删除

[JXHttpTool POST:@"deleteErrors" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inTestId":\_currentModel.TestId?:EMPTY\_STRING} success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

[self.navigationController popViewControllerAnimated:YES];

} failure:nil];

};

rightItem.clickRightBlock = ^{

//点击更多

[self showList];

};

self.navigationItem.rightBarButtonItem = [[UIBarButtonItem alloc] initWithCustomView:rightItem];

} else if (self.QuestionStuComeIn || self.HomeWorkStuComeIn) {

//自我测试

[JXHttpTool POST:@"beginDateTest" parameters:@{@"inId":self.inId?:EMPTY\_STRING,@"inClassId":self.ClassId?:EMPTY\_STRING,@"inStudentId":self.StudentId?:EMPTY\_STRING} success:^(id responseObject) {

} failure:nil];

TestMySelfLeftItem \*leftItem = [TestMySelfLeftItem viewFromNIB];

if (self.HomeWorkStuComeIn) {

//练习

[leftItem.rightBtn setImage:IMG(@"general\_icon\_operat") forState:UIControlStateNormal];

leftItem.clickLeftBlock = ^{

[UIAlertTool showAlertViewWithTitle:@"提示" message:@"您确定暂停做作业吗？" cancelButtonTitle:@"取消" otherButtonTitle:@"确定" confirm:^{

[self.navigationController popViewControllerAnimated:YES];

} cancle:nil];

};

leftItem.clickRightBlock = ^{

PaperListVC \*listVC = [[PaperListVC alloc] init];

listVC.refreshPageBlock = ^(int index) {

\_currentPage = index;

[self.scrollView scrollRectToVisible:CGRectMake(\_currentPage \* SCREEN\_WIDTH, 0, SCREEN\_WIDTH, SCREEN\_HEIGHT) animated:NO];

\_currentModel = \_allDatasArr[\_currentPage];

};

listVC.isBrowerPaperList = YES;

listVC.inId = self.inId?[self.inId numberValue]:@0;

listVC.answerDict = self.answerDict;

// listVC.inClassId = model.ClassId;

// listVC.inStudentId = model.StudentId;

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

};

} else {

//自我测试

leftItem.clickLeftBlock = ^{

[UIAlertTool showAlertViewWithTitle:@"提示" message:@"您确定暂停自我测试吗？" cancelButtonTitle:@"取消" otherButtonTitle:@"确定" confirm:^{

[self.navigationController popToRootViewControllerAnimated:YES];

} cancle:nil];

};

leftItem.clickRightBlock = ^{

[self.navigationController popViewControllerAnimated:YES];

};

}

self.navigationItem.leftBarButtonItem = [[UIBarButtonItem alloc] initWithCustomView:leftItem];

TestMySelfRightItem \*rightItem = [TestMySelfRightItem viewFromNIB];

rightItem.clickRightBlock = ^{

//点击更多

[self showList];

};

self.navigationItem.rightBarButtonItem = [[UIBarButtonItem alloc] initWithCustomView:rightItem];

} else {

self.navigationItem.rightBarButtonItem = [UIBarButtonItem barButtonItemWithImage:@"general\_icon\_more" andBarButtonItemBlock:^{

[self showList];

}];

}

self.scrollView = [[UIScrollView alloc] initWithFrame:self.view.bounds];

self.scrollView.delegate = self;

self.scrollView.pagingEnabled = YES;

[self.view addSubview:self.scrollView];

[self reqeustAllData];

[DefaultCenter addObserver:self selector:@selector(willHideNotification) name:UIKeyboardWillHideNotification object:nil];

}

-(void)viewWillAppear:(BOOL)animated

{

[super viewWillAppear:animated];

if (self.QuestionStuComeIn

|| self.HomeWorkStuComeIn) {

//自我测试

if (!self.timer) {

self.timer = [NSTimer scheduledTimerWithTimeInterval:1.0f target:self selector:@selector(refreshTimer:) userInfo:nil repeats:YES];

}

}

[IQKeyboardManager sharedManager].shouldResignOnTouchOutside = YES;

}

-(void)viewWillDisappear:(BOOL)animated

{

[super viewWillDisappear:animated];

if (self.timer) {

[self.timer invalidate];

self.timer = nil;

}

}

- (void)refreshTimer:(NSTimer \*)timer

{

\_allTime ++;

TestMySelfRightItem \*rightItem = self.navigationItem.rightBarButtonItem.customView;

rightItem.time.text = [DJUtils getMMSSFromSS:[NSString stringWithFormat:@"%d",\_allTime]];

}

- (void)collectPaper

{

PaperBrowseCollectListView \*listView = [PaperBrowseCollectListView viewFromNIB];

listView.clickCommitBlock = ^(NSNumber \*inFavId) {

//点击提交收藏

[JXHttpTool POST:@"addFavorites" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inFavId":inFavId?:EMPTY\_STRING,@"inTestId":\_currentModel.Id?:EMPTY\_STRING} success:^(id responseObject) {

\_currentModel.IsFavo = @1;

[UISlipUpTool disMiss];

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

} failure:nil];

};

listView.clickCreatCollectBlock = ^{

//点击创建收藏夹

[UISlipUpTool disMiss];

HomeCollectCreateAndEdit \*collectionVC = kStoryboardVC(@"Home", @"HomeCollectCreateAndEditSBID");

collectionVC.isAddCollect = YES;

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

};

[UISlipUpTool slipUpWithView:listView andSpeed:0.5];

}

- (void)cancelCollectPaper

{

NSLog(@"%@",\_currentModel);

[JXHttpTool POST:@"cancelFavorites" parameters:@{@"inPlatformId":KACCOUNTTOOLOUTUSER.PlatformNumber,@"inFavId":\_currentModel.FavId?:EMPTY\_STRING,@"inTestId":\_currentModel.TestId?:EMPTY\_STRING} success:^(id responseObject) {

SHOWSUCCESSAFTERONESECOND(responseObject[MSG]);

\_currentModel.IsFavo = @0;

if (self.inFavId) {

[self.navigationController popViewControllerAnimated:YES];

}

} failure:nil];

}

- (void)showList {

NSArray \*images;

if (![\_currentModel.IsFavo isEqualToNumber:@0]) {

//显示取消收藏

images = @[@"取消收藏", @"纠错",@"分享"];

} else {

//显示收藏

images = @[@"收藏", @"纠错",@"分享"];

}

[FTPopOverMenu showFromSenderFrame:CGRectMake(self.view.frame.size.width - 40, 20, 40, 40)

withMenuArray:images

imageArray:nil

doneBlock:^(NSInteger selectedIndex) {

if (selectedIndex == 0) {

//收藏

if (![\_currentModel.IsFavo isEqualToNumber:@0]) {

//显示取消收藏

[self cancelCollectPaper];

} else {

//显示收藏

[self collectPaper];

}

} else if (selectedIndex == 1) {

//我要纠错

NSString \*inTestId;

if (self.inChapterId || self.inKnowledgeId) {

inTestId = [NSString stringWithFormat:@"%@",\_currentModel.Id?:EMPTY\_STRING];

} else {

inTestId = [NSString stringWithFormat:@"%@",\_currentModel.TestId?:EMPTY\_STRING];

}

PaperCorrectMistakeTVC \*mistakeVC = kStoryboardVC(@"Paper", @"PaperCorrectMistakeTVCSBID");

mistakeVC.inTestId = inTestId;

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

} else if (selectedIndex == 2) {

//分享

NSString \*url = [NSString stringWithFormat:@"%@%@?inId=%@",kBaseURL,@"/TestInfoShow",self.inId?:EMPTY\_STRING];

[SnShareTool showShareWithImage:@"logo" url:url andTitle:@"因卓教育平台题目"];

}

NSLog(@"done block. do something. selectedIndex : %ld", (long)selectedIndex);

} dismissBlock:^{

NSLog(@"user canceled. do nothing.");

}];

}

- (void)reqeustAllData

{

if (self.inFavId) {

//收藏列表进入

[self requestCollectData];

} else if (self.selectPaperShowModels) {

//选择试题进入

\_allDatasArr = @[].mutableCopy;

for (int j = 0; j < self.selectPaperShowModels.count; j ++) {

PaperBrowseModel \*model = self.selectPaperShowModels[j];

[\_allDatasArr addObject:model];;

[self refreshDataWith:model andCount:j canTouch:YES];

}

\_currentModel = \_allDatasArr[\_currentPage];

} else if (self.muDict) {

//错题本列表进入

#warning 接口有问题

[JXHttpTool POST:@"getErrorsList" parameters:self.muDict success:^(id responseObject) {

PaperBrowseModel \*model = [PaperBrowseModel mj\_objectWithKeyValues:responseObject[@"outList"][0]];

[\_allDatasArr addObject:model];

\_currentModel = \_allDatasArr[0];

[self refreshDataWith:model andCount:0 canTouch:YES];

} failure:nil];

} else if (self.QuestionStuComeIn) {

//点击题 进入 自我测试

#warning 答题 待写

NSDictionary \*dict;

if (self.inChapterId) {

//选择的章节点

dict = @{@"inChapterId":self.inChapterId?:EMPTY\_STRING};

} else if (self.inKnowledgeId) {

//选择的知识点

dict = @{@"inKnowledgeId":self.inKnowledgeId?:EMPTY\_STRING};

}

[JXHttpTool POST:@"getappQuestionsList" parameters:dict success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

return ;

}

\_allDatasArr = @[].mutableCopy;

for (int i = 0 ; i < [responseObject[@"outList"] count]; i ++) {

PaperBrowseModel \*model = [PaperBrowseModel mj\_objectWithKeyValues:responseObject[@"outList"][i]];

[\_allDatasArr addObject:model];

[self refreshDataWith:model andCount:i canTouch:YES];

}

\_currentModel = \_allDatasArr[\_currentPage];

} failure:nil];

} else if (self.HomeWorkStuComeIn){

//做作业进入 ,

[JXHttpTool POST:@"testPaperTaskOne" parameters:@{@"inId":self.inId?:EMPTY\_STRING,@"incurrentPage":@(\_homeWordCurrentPage)} success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

return ;

}

for (int i = 0 ; i < [responseObject[@"outList"] count]; i ++) {

PaperBrowseModel \*model = [PaperBrowseModel mj\_objectWithKeyValues:responseObject[@"outList"][i]];

[\_allDatasArr addObject:model];

}

if (![responseObject[@"Page"][@"CurrentPage"] isEqualToNumber:responseObject[@"Page"][@"TotalPage"] ]) {

//最后一页的时候

\_homeWordCurrentPage ++;

[self reqeustAllData];

return;

}

for (int j = 0; j < \_allDatasArr.count; j ++) {

PaperBrowseModel \*model = \_allDatasArr[j];

[self refreshDataWith:model andCount:j canTouch:YES];

}

\_currentModel = \_allDatasArr[\_currentPage];

} failure:nil];

} else if (self.paperBrowerListComeIn){

//点击试题列表 大圈圈 进入

if (self.numbers) {

SunOrderArray \*postOrderArray = [[SunOrderArray alloc] init];

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

[JXHttpTool POST:@"testPaperTaskOne" parameters:@{@"inId":self.inId?:EMPTY\_STRING,@"incurrentPage":self.numbers[i]?:EMPTY\_STRING} success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

return ;

}

[postOrderArray addArray:@[[PaperBrowseModel mj\_objectWithKeyValues:responseObject[@"outList"][0]]] WithIndex:i];

if (self.numbers.count == [postOrderArray allDatas].count) {

//便利完之后

\_allDatasArr = @[].mutableCopy;

for (int j = 0 ; j < [[postOrderArray allDatas] count]; j ++) {

PaperBrowseModel \*model = [PaperBrowseModel mj\_objectWithKeyValues:[postOrderArray allDatas][j]];

[\_allDatasArr addObject:model];

[self refreshDataWith:model andCount:j canTouch:NO];

}

\_currentPage = self.currentPageUI;

[self.scrollView scrollRectToVisible:CGRectMake(\_currentPage \* SCREEN\_WIDTH, 0, SCREEN\_WIDTH, SCREEN\_HEIGHT) animated:NO];

\_currentModel = \_allDatasArr[\_currentPage];

}

} failure:nil];

}

}

} else {

//选择范围进入

[self requestQuestionData];

}

}

- (void)requestCollectData

{

//试题列表

[JXHttpTool POST:@"getFaQestionList" parameters:@{@"inPlatformId":self.inPlatformId?:EMPTY\_STRING,@"inFavId":self.inFavId?:EMPTY\_STRING,@"inshowCount":@"5",@"incurrentPage":@"1"} success:^(id responseObject) {

SunOrderArray \*postOrderArray = [[SunOrderArray alloc] init];

for (int i = 0; i < [self.allNum integerValue]; i ++) {

[JXHttpTool POST:@"getFaQestionList" parameters:@{@"inPlatformId":self.inPlatformId?:EMPTY\_STRING,@"inFavId":self.inFavId?:EMPTY\_STRING,@"inshowCount":@"1",@"incurrentPage":[NSString stringWithFormat:@"%d",i + 1]} success:^(id responseObj) {

if ([responseObj[@"outList"] count] == 0) {

return ;

}

[postOrderArray addArray:@[[PaperBrowseModel mj\_objectWithKeyValues:responseObj[@"outList"][0]]] WithIndex:i];

if ([self.allNum integerValue] == [postOrderArray allDatas].count) {

//便利完之后

\_allDatasArr = @[].mutableCopy;

for (int j = 0; j < [postOrderArray allDatas].count; j ++) {

PaperBrowseModel \*model = [postOrderArray allDatas][j];

[\_allDatasArr addObject:model];;

[self refreshDataWith:model andCount:j canTouch:YES];

}

\_currentModel = \_allDatasArr[\_currentPage];

}

} failure:nil];

}

} failure:nil];

}

- (void)requestQuestionData

{

NSDictionary \*dict;

if (self.inChapterId) {

//选择的章节点

dict = @{@"inChapterId":self.inChapterId?:EMPTY\_STRING,@"incurrentPage":@"1",@"inshowCount":@"5"};

} else if (self.inKnowledgeId) {

//选择的知识点

dict = @{@"inKnowledgeId":self.inKnowledgeId?:EMPTY\_STRING,@"incurrentPage":@"1",@"inshowCount":@"5"};

}

[JXHttpTool POST:@"getPageQuestionsList" parameters:dict success:^(id responseObject) {

if ([responseObject[@"outList"] count] == 0) {

return ;

}

\_allDatasArr = @[].mutableCopy;

for (int i = 0 ; i < [responseObject[@"outList"] count]; i ++) {

PaperBrowseModel \*model = [PaperBrowseModel mj\_objectWithKeyValues:responseObject[@"outList"][i]];

[\_allDatasArr addObject:model];

[self refreshDataWith:model andCount:i canTouch:self.paperBrowerListComeIn?NO:YES];

}

\_currentModel = \_allDatasArr[\_currentPage];

} failure:nil];

}

- (void)refreshDataWith:(PaperBrowseModel \*)model andCount:(int)count canTouch:(BOOL)canTouch

{

//10道题

self.scrollView.contentSize = CGSizeMake((count + 1) \* SCREEN\_WIDTH, SCREEN\_HEIGHT);

//循环配置每道题的界面

BOOL isShow = NO;

if (self.HomeWorkStuComeIn) {

if (count == \_allDatasArr.count - 1) {

//最后一页的话

//做作业显示提交按钮

isShow = YES;

}

}

PaperObjectView \*objView = [[PaperObjectView alloc] initWithFrame:CGRectMake(count \* SCREEN\_WIDTH, 0, SCREEN\_WIDTH, SCREEN\_HEIGHT) andDatas:model allNum:0 isErrorComeIn:self.muDict?YES:NO showCommitBtn:isShow andAnswerDict:self.answerDict canTouch:canTouch];

if (self.muDict) {

ErrorComeDetailRightItem \*item = (ErrorComeDetailRightItem \*) self.navigationItem.rightBarButtonItem.customView;

if ([model.IsConquer isEqualToNumber:@1]) {

item.leftBtn.selected = YES;

} else {

item.leftBtn.selected = NO;

}

}

objView.clickCommitBlock = ^(NSMutableDictionary \*insertStrings) {

[self commitAnswer];

};

[self.scrollView addSubview:objView];

}

- (void)commitAnswer

{

[JXHttpTool upLoadCommitDataWithUrlString:@"subjectiveSubmit" parameters:self.answerDict completionHandler:^(id responseObject, NSError \*error) {

if (responseObject) {

if (![responseObject[@"result"] isEqualToString:@"00"]) {

if (responseObject[@"msg"] && ![responseObject[@"msg"] isEqualToString:EMPTY\_STRING]) {

if ([responseObject[@"msg"] isEqualToString:@"非法小问!"]) {

SHOWERRORAFTERONESECOND(@"答案不能为空");

} else {

SHOWERRORAFTERONESECOND(responseObject[@"msg"]);

}

}

return ;

}

SHOWSUCCESSAFTERONESECOND(responseObject[@"msg"]);

[self.navigationController popToRootViewControllerAnimated:YES];

}

}];

}

#pragma uiscrollview delegate

-(void)scrollViewDidScroll:(UIScrollView \*)scrollView

{

\_currentPage = self.scrollView.contentOffset.x/SCREEN\_WIDTH;

\_currentModel = \_allDatasArr[\_currentPage];

NSLog(@"啊啊啊啊啊啊啊%ld",(long)\_currentPage);

NSLog(@"%@",\_allSuccessPro);

}

-(void)scrollViewWillBeginDragging:(UIScrollView \*)scrollView

{

[DefaultCenter postNotificationName:UIKeyboardWillHideNotification object:nil];

}

- (void)willHideNotification

{

NSLog(@"%ld",(long)\_currentPage);

if (\_currentModel.outProList.count != 0) {

if ([[\_currentModel.outProList[0] QuType] isEqualToString:@"填空题"]) {

// for (UIView \*view in self.scrollView.subviews) {

// if ([view isKindOfClass:[PaperObjectView class]]) {

if ([self.scrollView.subviews[\_currentPage + 1] isKindOfClass:[PaperObjectView class]]) {

PaperObjectView \*paper = (PaperObjectView \*)self.scrollView.subviews[\_currentPage + 1];

if (paper.insertTypeView) {

//如果有填空题 就保存所有答案

for (NSString \*str in [paper.insertTypeView getInsertArr]) {

if (![str isEqualToString:EMPTY\_STRING]) {

BOOL havaed = NO;

// NSMutableArray \*muarr = [NSMutableArray arrayWithArray:self.answerDict[@"inAnswerObj"]];

for (NSMutableDictionary \*dict in self.answerDict[@"inAnswerObj"]) {

if ([dict[@"ProId"] isEqualToNumber:((PaperBrowseOptionModel \*)\_currentModel.outProList[0]).Id]) {

dict[@"Answer"] = [paper.insertTypeView getInsertStrings][@"Answer"];

havaed = YES;

break;

}

}

if (!havaed) {

[self.answerDict[@"inAnswerObj"] addObject:[paper.insertTypeView getInsertStrings]];

}

return;

}

}

}

}

}

}

// NSLog(@"%@",self.answerDict);

// }

// }

}

-(void)scrollViewWillEndDragging:(UIScrollView \*)scrollView withVelocity:(CGPoint)velocity targetContentOffset:(inout CGPoint \*)targetContentOffset

{

if (!self.QuestionStuComeIn) {

return;

}

if(velocity.x>0){

//右滑

NSLog(@"右滑");

if ([self.answerDict[@"inAnswerObj"] count] != 0) {

BOOL have = NO;

for (NSMutableDictionary \*dict in self.answerDict[@"inAnswerObj"]) {

if ([dict[@"TestId"] isEqualToNumber:\_currentModel.Id]) {

[dict removeObjectForKey:@"TestId"];

have = YES;

[JXHttpTool POST:@"oneTestCorrecting" parameters:@{@"inPros":@[dict].mj\_JSONString} success:^(id responseObject) {

[\_allSuccessPro setObject:responseObject[@"outList"][0][@"IsCorrect"] forKey:[NSString stringWithFormat:@"%@.%ld",\_currentModel.Id?:EMPTY\_STRING,(long)\_currentPage]];

[self pushListVC];

} failure:nil];

}

}

if (!have) {

[\_allSuccessPro setObject:@"-1" forKey:[NSString stringWithFormat:@"%@.%ld",\_currentModel.Id?:EMPTY\_STRING,(long)\_currentPage]];

[self pushListVC];

}

} else {

[\_allSuccessPro setObject:@"-1" forKey:[NSString stringWithFormat:@"%@.%ld",\_currentModel.Id?:EMPTY\_STRING,(long)\_currentPage]];

[self pushListVC];

}

}else{

//左滑

NSLog(@"左滑");

}

}

- (void)pushListVC

{

if (\_currentPage == \_allDatasArr.count - 1) {

PaperListVC \*paperList = [[PaperListVC alloc] init];

paperList.isTestCommitComeIn = YES;

paperList.allSuccessProDict = \_allSuccessPro;

TestMySelfRightItem \*rightItem = self.navigationItem.rightBarButtonItem.customView;

paperList.timeStr = rightItem.time.text;

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

}

}

@end