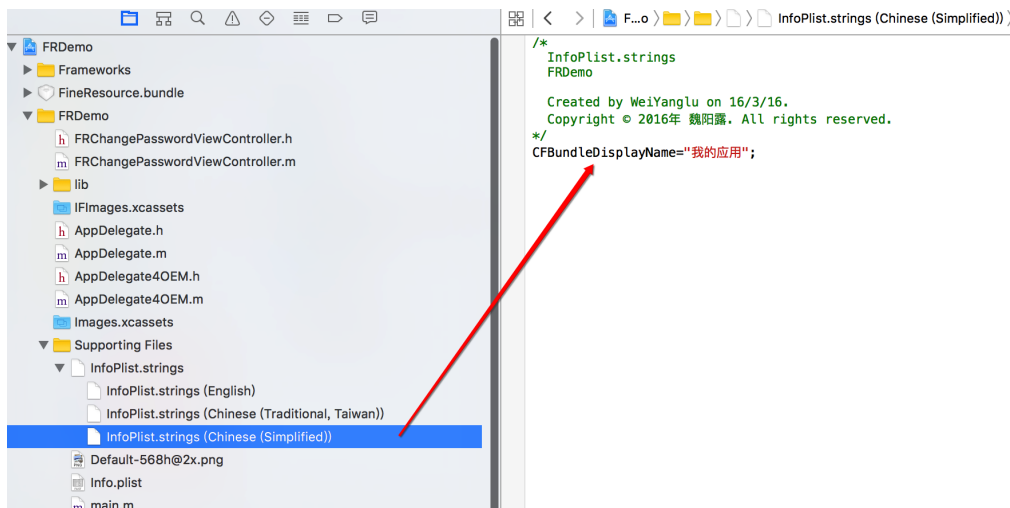


- 1.app
- 2.app
- 3.
- 4. FR
- 5.
- 6. App
- 7.
- 8. OEMApp
- 9.
- 10. App
- 11. App
 - "XGPush"
 - App
 -
 -
- 12.
- 13.
- 14.
- 15. APP
- 16. App
- 17.
- 18.app"FRDemo""
- 19.OEM

1.app

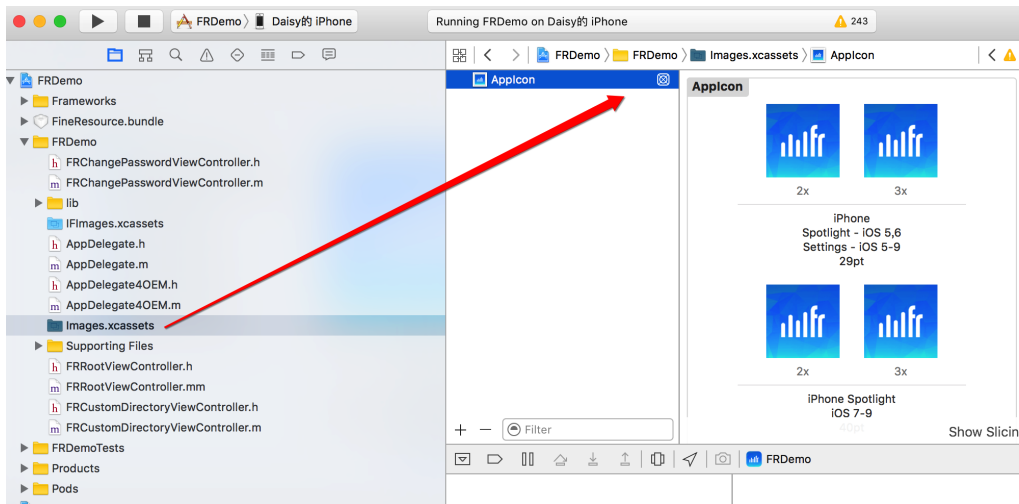
infoPlist.stringsCFBundleDisplayName()



2.app

:images.xcassets,appicon2x,3x

(ps:xcode2x,3x:https://developer.apple.com/library/ios/qa/qa1686/_index.html)



3.

```
[IFEntryViewController initWithPath:(NSString *)reportPath serverUrl:(NSString *)serverUrl viewType:
(IFEntryViewType)viewType parameters:(NSDictionary *) params];
```

reportPath	@"/mobile.test.cpt"	
serverUrl		
viewType	IFEntryViewTypePage IFEntryViewTypeWrite nil	
params		

4. FR

```
[IFIntegrationUtils logInto:DEFAULT_SERVERNAME serverUrl:url withUsername:username andPassword:password success:
^{
    //
    [IFIntegrationUtils loadReportTree:^(NSArray *reportsArray) {
        //
    } failure:^(NSString *msg) {
    } isObj:YES];
} failure:^(NSString *msg) {
    //
};
```

```
reportsArray
reportsArrayIFEntryNode
IFEntryNode[node isFolder]
node[node children]IFEntryNode
node[[IFEntryViewController alloc] initWithEntry:node]
```

5.

```
//
[IFIntegrationUtils loginInto:DEFAULT_SERVERNAME serverUrl:url withUsername:username andPassword:password success:
^{
    //
    [IFIntegrationUtils favorites:^(NSMutableArray *favorites) {
        //
    } failure:^(NSString *message) {
    }
}];
} failure:^(NSString *msg) {
}];
```

IFEntryNode

6. App

iOS

7.

```
[IFIntegrationUtils logout];
```

8. OEMApp

a. OEMAppApp

b. App

1.FR

```
[IFOEMUtils removeDemoServer];
```

2.

```
[IFOEMUtils addServerWithName:(NSString *)serverName andURL:(NSString *)serverURL addUsername:(NSString *)
username addPassword:(NSString *)password userEditable:(BOOL)userEditable];
```

serverName		
serverURL		
username		
password		
userEditable		

c.

1.FR

```
[IFOEMUtils removeAbout];
```

2.

```
[IFOEMUtils setAppVersion:(NSString *) version];//
        [IFOEMUtils setCopyright:(NSString *) copyright];//
        [IFOEMUtils setAppIconName:(NSString *) iconName];//
```

d. AppFR/App

```
[IFOEMUtils setMessageHandler:(void (^) (NSString *message, BOOL isSuccess)) handler];
```

handler	

Block

message	
isSuccess	YESNO/

e. App

```
[IFOEMUtils setAppSettingBarButtonItem:(UIBarButtonItem *) settingItem];
```

settingItem

9.

AppDelegatedidFinishLaunchingWithOptions

```
[IFIntegrationUtils createTable];
```

8FR

```
IFFrameAppSettingViewController *settingViewController = [[IFFrameAppSettingViewController alloc]
initWithCustomItems:customItems];
```

customItems:IFFrameAppSettingItem

```
IFFrameAppSettingItem *logoutItem = [[IFFrameAppSettingItem alloc] initWithTitle:@" " andAction:^
(UITableViewController *selectedCell) {
    [IFIntegrationUtils logOut];
    [self dismissViewControllerAnimated:YES completion:nil];
}];
NSArray *logOutSection = [[NSArray alloc] initWithObjects:logoutItem, nil];
NSArray *customItems = [[NSArray alloc] initWithObjects:logOutSection, nil];
IFFrameAppSettingViewController *settingViewController = [[IFFrameAppSettingViewController alloc]
initWithCustomItems:customItems];
```

10. App

http/dap

```
//
[IFIntegrationUtils loginInto:DEFAULT_SERVERNAME serverUrl:url withUsername:username andPassword:password success:^{
    //
    if([IFIntegrationUtils canChangePassword]) {
        //22originalPasswordnewPassword
        [IFIntegrationUtils changePassword:originalPassword newPassword:newPassword success:^(
            //
            } failure:^(NSString *message) {
                //
            }];
    } else {
        //
    }
} failure:^(NSString *msg) {
    //
};
};
```

11. App

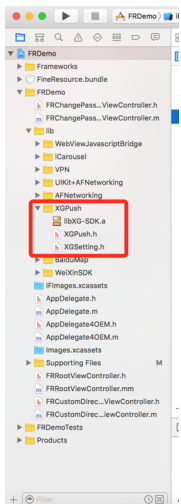
App

AppACCESS IDACCESS KEYSECRET KEY

ACCESSIDACCESSKEY

"XGPush"

FRDemococoapodscocoapods



App

```
//AppDelegate.m
//Appaccess idaccess key()
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    [IFOEMUUtils setXinGeAppId:123456788]; //access id
    [IFOEMUUtils setXinGeAppKey:@"ISHGJ5A7342D"]; //accesskey
    //do somethigs
    return YES;
}
```

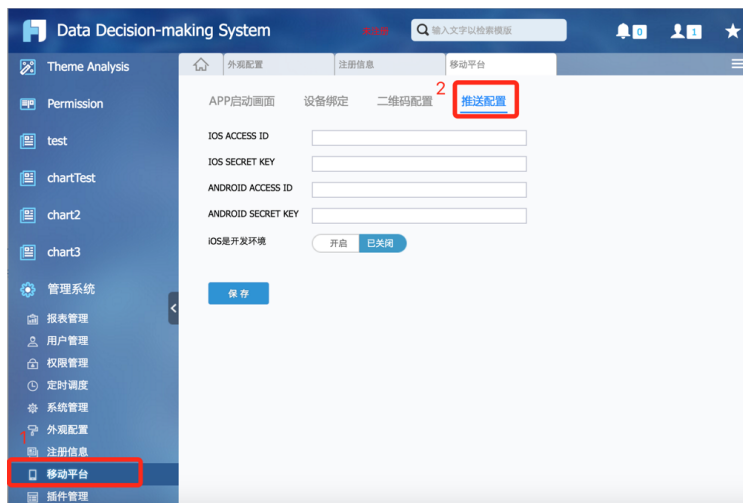
OEMAppDelegate

```

- (void)application:(UIApplication *)application didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)
deviceToken {
    NSString * deviceTokenStr = [XGPush registerDevice:deviceToken];
    [XGPush registerDevice:deviceToken];
    [IFIntegrationUtils setDeviceToken:deviceTokenStr];
}
- (void)application:(UIApplication *)application didRegisterUserNotificationSettings:
(UIUserNotificationSettings *)notificationSettings
{
    [application registerForRemoteNotifications];
}
- (void)application:(UIApplication *)application didReceiveRemoteNotification:(NSDictionary *)userInfo{
    //
}

```

FineReport



iOSACCESS ID secret key

a. homeapp

b.



添加任务

基本属性 > 调度周期 > 输出设置

任务名称: test

模板名称: GettingStarted.cpt

类型: ☒ 预览 ☐ 填报 ☐ 分析

参数设置

名称	类型	值
----	----	---

下一步 取消

添加任务

基本属性 > 调度周期 > 输出设置

开始时间: ☒ 配置结束立即执行 ☐ 设定开始时间 2016-10-11 15:50:52

执行频率: ☒ 只执行一次 ☐ 简单重复执行 ☐ 明细频率设置 ☐ 表达式设定

结束时间: ☒ 任务执行一次后即结束

执行条件: ☒ 始终执行 ☐ 公式判断

上一步 下一步 取消

添加任务

基本属性 > 调度周期 > 输出设置

文件名称: testaaaaa

文件描述:

挂载虚拟目录: Theme Analysis

☐ 根据用户选择不同的内容

通知与存档: ☐ excel ☐ word ☐ pdf ☒ cpr

☐ 邮件通知

☐ FTP上传附件

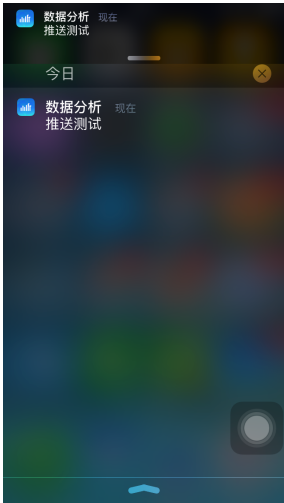
☐ 自定义附件处理

☐ 打印文件

☒ 推送消息

消息内容: 推送测试

上一步 提交 取消



12.

IFIntegrationUtils loadReportTreeIFEntryNodeentryNode

```
[IFIntegrationUtils readEntryImage:entryNode.entryId coverId:entryNode.coverId sucess:^(UIImage *image) {
    //image
}];
```

13.

/

```
IFEntryView *entryView = [[IFEntryView alloc] initWithPath:@"app/DetailedDrillA-phone.cpt" serverUrl:@"http://www.finereporthelp.com:8889/app/ReportServer" viewType:IFEntryViewTypePage parameters:nil];
[entryView setFrame:CGRectMake(0, 64, 800, 600)];
entryView.entryViewDelegate = self;
[self.view addSubview:entryView];
[entryView doLoad];
```

IFEntryViewDelegate

```
-(void)pushHyperLink:(UIViewController *)viewController withAnimate:(BOOL)animate {
    //View
}
```

14.

1.supporting filesLocalizable.strings(demo(chinese simple,chinese traditional,english)) :http://blog.sina.com.cn/s/blog_7b9d64af0101jncz.htmlios.

2.key:

```
NSString * myKey = NSLocalizedString(@"MY_KEY", nil);
```

```
myKeykey"MYKEY".Localizable.strings(english)"MYKEY"="my key";Localizable.strings(chinese(simple))"MYKEY"="";key myKey"MYKEY"
```

15. APP

supporting filesInfoPlist.strings(New File——Resource——String FileInfoPlist.strings)LocalizationKeyCFBundleDisplayName

16. App

url schemeurl schemeFineReportcom.fr.FineReport

FRAApp

```
//url//URLencode[string stringByAddingPercentEscapesUsingEncoding:NSUTF8StringEncoding]
NSURL *url = [NSURL URLWithString:@"FineReport://"];
//NSURL *url = [NSURL URLWithString:@"com.fr.FineReport://"]; //
//NSURL *url = [NSURL URLWithString:@"com.fr.FineReport://?
serverName=test&serverUrl=xxxxxx&fr_username=a&fr_password=123"]; //, urlencode
//JSON
//NSURL *url = [NSURL URLWithString:@"com.fr.FineReport://{\"serverName\": \"xxxxx\", \"serverUrl\": \"
http://xxxxx\", \"fr_username\": \"a\", \"fr_password\": \"123\"}"]
//
//1. serverNameserverUrl
//2. serverName, serverUrl""
//3. serverName,

[[UIApplication sharedApplication] openURL:url];
```

iOS9AppFineReportcom.fr.FineReportscheme

Bundle Version	String	URL
▼ LSAApplicationQueriesSchemes	Array	(2 items)
Item 0	String	FineReport
Item 1	String	com.fr.FineReport

17.

16

▼ LSAApplicationQueriesSchemes	Array	(2 items)
Item 0	String	wechat
Item 1	String	weixin

18.app“FRDemo”“”

appidappid

appid

▼ URL Types (1)

weixin

No image specified

Identifier weixin

Icon None

URL Schemes wx838c58da5e8d727e

Role Editor

► Additional url type properties (0)

+

appappid

```

13
14 - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
15     [IFOEMUtils setCopyright:@"test"];
16     [IFOEMUtils removeDemoServer];
17     [IFOEMUtils addServerWithName:@"MyServer" andURL:@"http://www.finereporthehelp.com:8889/app/ReportServer"
18         addUsername:@"demo" addPassword:@"demo"];
19
20     uint32_t appId = 2200157433;
21     [IFOEMUtils setXinGeAppId:appId];
22     [IFOEMUtils setXinGeAppKey:@"ISHGJ5A7342D"];
23     [IFOEMUtils setWeiXinAppID:@"wx838c58da5e8d727e"];
24
25     [super application:application didFinishLaunchingWithOptions:launchOptions];
26     return YES;
27 }

```

19.OEM

```

UIBarButtonItem *setting = [[UIBarButtonItem alloc] initWithTitle:@"" style:UIBarButtonItemStylePlain target:
self action:@selector(setting)];

[IFOEMUtils setAppSettingBarButtonItem:setting];

```

FR

```
- (void) setting {
    //
    IFFrameAppSettingItem *logoutItem = [[IFFrameAppSettingItem alloc] initWithTitle:@"\" andAction:^
(UITableViewCell *selectedCell) {
    //
    [IFIntegrationUtils logout];
    //
    [[self topViewControllerWithRootViewController:[UIApplication sharedApplication].delegate.window.
rootViewController] dismissViewControllerAnimated:YES completion:nil];
    }];

    //
    NSArray *logoutSection = [[NSArray alloc] initWithObjects:logoutItem, nil];
    NSArray *customItems = [[NSArray alloc] initWithObjects:logoutSection, nil];

    //
    IFFrameAppSettingViewController *settingViewController; = [[IFFrameAppSettingViewController alloc]
initWithCustomItems:customItems];

    ///Pad
    BOOL isPhone = [[UIDevice currentDevice] userInterfaceIdiom] == UIUserInterfaceIdiomPhone;
    UIViewController *topViewController = [self topViewControllerWithRootViewController:[UIApplication
sharedApplication].delegate.window.rootViewController];
    if(isPhone) {
        [topViewController.navigationController pushViewController:settingViewController animated:YES];
    } else {
        UINavigationController *navigationController = [[UINavigationController alloc] initWithRootViewController:
settingViewController];
        navigationController.modalPresentationStyle = UIModalPresentationFormSheet;
        [topViewController presentViewController:navigationController animated:YES completion:nil];
    }
}
//
- (UIViewController*)topViewControllerWithRootViewController:(UIViewController*)rootViewController {
    if ([rootViewController isKindOfClass:[UITabBarController class]]) {
        UITabBarController* tabBarController = (UITabBarController*)rootViewController;
        return [self topViewControllerWithRootViewController:tabBarController.selectedViewController];
    } else if ([rootViewController isKindOfClass:[UINavigationController class]]) {
        UINavigationController* nav = (UINavigationController*)rootViewController;
        return [self topViewControllerWithRootViewController:nav.topViewController];
    } else if (rootViewController.presentedViewController) {
        UIViewController* presentedViewController = rootViewController.presentedViewController;
        if([presentedViewController isKindOfClass:[UIAlertController class]]) {
            return rootViewController;
        }
        return [self topViewControllerWithRootViewController:presentedViewController];
    } else {
        return rootViewController;
    }
}
```

