



iOS 各种传值方式

收藏人: cccshq

2014-04-10 | 阅: 2813 转: 36 | 来源 [A] [图] | 分享 ▼

属性传值 将A页面所拥有的信息通过属性传递到B页面使用
B页面定义了一个naviTitle属性，在A页面中直接通过属性赋值将A页面中的值传到B页面。

A页面DetailViewController.h文件

```
#import <UIKit/UIKit.h>
#import "DetailViewController.h"
@interface RootViewController :UIViewController<ChangeDelegate>
{
    UITextField *tf;
}
@end
```

A RootViewController.m页面实现文件

```
#import "RootViewController.h"
#import "DetailViewController.h"

@interface RootViewController ()
@end
@implementation RootViewController
//核心代码
-(void)loadView
{
    UIButton *btn = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    btn.frame = CGRectMake(0, 0, 100, 30);
    [btn setTitle:@"Push" forState:0];
    [btn addTarget:self action:@selector(pushAction:) forControlEvents:UIControlEventTouchUpInside];
    [self.view addSubview:btn];
}

-(void)pushAction:(id)sender
{
    tf = (UITextField *) [self.viewviewWithTag:1000];
    //导航push到下一个页面
    //pushViewController 入栈引用计数+1,且控制权归系统

    DetailViewController *detailViewController = [[DetailViewController alloc] init];
    //属性传值,直接属性赋值
    detailViewController.naviTitle = tf.text;
    //导航push到下一个页面
    [self.navigationController pushViewController:detailViewController animated:YES];
    [detailViewController release];
}
}
```

B页面DetailViewController.h文件

```
#import <UIKit/UIKit.h>
@interface DetailViewController :UIViewController
{
    UITextField *textField;
    NSString *_naviTitle;
}
@property(n nonatomic, retain) NSString *naviTitle;
@end
```

B页面.m实现文件

```
#import "DetailViewController.h"
@interface DetailViewController ()
@end
@implementation DetailViewController
@synthesize naviTitle = _naviTitle;
-(void)loadView
{
    self.view = [[UIView alloc] initWithFrame:CGRectMake(0,0, 320,480)]autorelease];
    self.title = self.naviTitle ;
}
}
```

代理传值

A页面push到B页面，如果B页面的信息想回传（回调）到A页面，用用代理传值，其中B定义协议和声明代理，A确认并实现代理，A作为B的代理

ios开发工程师

数据库学习 手机ap

最新文章

- UITableView的基本使用二（性
- IOS NSString类型十六进制转换
- IOS中十六进制的颜色转换为UI
- isKindOfClass isMemberOfClas
- (译) In App Purchases 入门
- IOS本地,APNS远程推送（具体

缤特力

文胸	女包
婚纱	凉席
凉鞋	座垫
童装	长T恤
小米3	短T恤 ¥4

热门文章

- 解放军35名现役上将名单
- 谁是中国最伟大的民营跨国企业
- 熟记太极要诀,让您终生受益
- 哦,你是我的父亲
- 绝佳美少妇
- ?你所不知道的东莞: 沉默的大?
- 各种图形计算公式大全
- 视频教育导航 高一化学 (全33集)
- 【生活经】厨房清洁小技巧 让你
- 读书知真趣: 美句品读!
- 不容错过, 国学大师推荐书目下
- 网络上很猛的语句, 笑到气不

供应新大陆电子猫眼|
郑州新大陆智能科技有限公司

供应新大陆电子猫眼|郑州新大陆智
能科技有限公
司
河南省 郑州市
更多电子猫眼 价格 图片

- 1 美国购房移民 开放啦
- 2 头皮屑大块, 头皮瘙痒, 一次见效
- 3 上海启本 带给您节能绿色生活
- 4 停车收费系统专业, 诚信, 快速
- 5 右脑记忆一提高100万倍记忆力!
- 6 8000起薪 it培训 学不会退费!
- 7 开什么店赚钱, 千元投资创业项
- 8 绿化我来秀, 话氧森林带回家!
- 9 在家做家庭煮妇?学手艺吧
- 10 手工活招聘, 月入30000.
- 11 电话监控, 了解对方在干么?
- 12 沪上最好喝的血糯米奶茶

```
#import <UIKit/UIKit.h>
#import "DetailViewController.h"
@interface RootViewController : UIViewController<ChangeDelegate>
{
    UITextField *tf;
}
@end

A页面RootViewController.m实现文件
#import "RootViewController.h"
#import "DetailViewController.h"

@interface RootViewController ()
@end
@implementation RootViewController
//核心代码
-(void)loadView
{
    UIButton *btn = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    btn.frame = CGRectMake(0, 0, 100, 30);
    [btn setTitle:@"Push" forState:0];
    //A页面push到B页面

    [btn addTarget:self action:@selector(pushAction:) forControlEvents:UIControlEventTouchUpInside];
    [self.view addSubview:btn];
}

-(void)pushAction:(id)sender
{
    tf = (UITextField *)[self.view viewWithTag:1000];
    //导航push到下一个页面
    //pushViewController 入栈引用计数+1, 且控制权归系统
    DetailViewController *detailViewController = [[DetailViewController alloc] init];
    //代理传值
    detailViewController.delegate = self; //让其自身作为代理人
    //导航push到下一个页面
    [self.navigationController pushViewController:detailViewController animated:YES];
    [detailViewController release];
}

//实现代理方法
-(void)changeTitle:(NSString *)aStr
{
    tf = (UITextField *)[self.view viewWithTag:1000];
    tf.text = aStr; //将从B页面传入的参数赋给A页面中的TextField
    tf.text = aStr;
}

B页面DetailViewController.m文件
#import <UIKit/UIKit.h>
@interface DetailViewController : UIViewController
{
    UITextField *textField;
    //定义代理
    id<ChangeDelegate> _delegate;
}
@property(nonatomic,assign) id<ChangeDelegate> delegate;
@end
//定义协议
@protocol ChangeDelegate <NSObject>
-(void)changeTitle:(NSString *)aStr; //协议方法
@end

B页面DetailViewController.h实现文件

#import "DetailViewController.h"
@interface DetailViewController ()
@end
@implementation DetailViewController
-(void)loadView
{
    self.view = [[[UIView alloc] initWithFrame:CGRectMake(0, 0, 320, 480)] autorelease];
    UIBarButtonItem *doneItem =
    [[UIBarButtonItem alloc] initWithBarButtonSystemItem:UIBarButtonSystemItemDone target:self action:@selector(doneAction)];
    self.navigationItem.rightBarButtonItem = doneItem;
    [doneItem release];
}

//pop回前一个页面
-(void)doneAction:(id)sender
{
    if (self.delegate && [self.delegate respondsToSelector:@selector(changeTitle:)]) //若代理存在且响应了changeTitle这个方法
    {
        // [self.delegate changeTitle:textField.text];
        [self.delegate changeTitle:textField.text]; //将textField.text参数传给changeTitle方法 让代理, 也就是A页面去实现这个方法
        NSLog(@"%@", self.navigationController.viewControllers);
        [self.navigationController popViewControllerAnimated:YES];
    }
}
```

单例传值（实现共享）

AppStatus.h 创建一个单例类 AppStatus

```
#import <Foundation/Foundation.h>

@interface AppStatus : NSObject
{
    NSString *_contextStr;
}
@property(nonatomic,retain)NSString *contextStr;

+(AppStatus *)shareInstance;

@end
```

AppStatus.m

```
#import "AppStatus.h"

@implementation AppStatus
@synthesize contextStr = _contextStr;

static AppStatus *_instance = nil;
+(AppStatus *)shareInstance
{
    if (_instance == nil)
    {
        _instance = [[super alloc] init];
    }
    return _instance;
}

-(id)init
{
    if (self = [super init])
    {
    }
    return self;
}

-(void)dealloc
{
    [super dealloc];
}

@end

A页面RootViewController.h

#import "RootViewController.h"
#import "DetailViewController.h"
#import "AppStatus.h"

@interface RootViewController ()

@end

@implementation RootViewController
-(void)loadView
{
    //核心代码
    UIButton *btn = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    btn.frame = CGRectMake(0, 0, 100, 30);
    [btn setTitle:@"Push" forState:0];
    [btn addTarget:self action:@selector(pushAction:) forControlEvents:UIControlEventTouchUpInside];
    [self.view addSubview:btn];
}

-(void)pushAction:(id)sender
{
    tf = (UITextField *)[self.view viewWithTag:1000];

    //单例传值 将要传递的信息存入单例中（共享中）
    // [[AppStatus shareInstance]setContextStr:tf.text]; 跟下面这种写法是等价的
    [AppStatus shareInstance].contextStr = tf.text;
    //导航push到下一个页面
    //pushViewController 入栈引用计数+1，且控制权归系统

    DetailViewController *detailViewController = [[DetailViewController alloc] init];

    //导航push到下一个页面
    [self.navigationController pushViewController:detailViewController animated:YES];
    [detailViewController release];
}
```

```
@end
```

```
B页面DetailViewController.h
```

```
#import <UIKit/UIKit.h>
```

```
@protocol ChangeDelegate;//通知编译器有此代理
```

```
@interface DetailViewController : UIViewController
```

```
{
```

```
    UITextField *textField;
```

```
}
```

```
@end
```

```
B页面DetailViewController.m
```

```
#import "DetailViewController.h"
```

```
#import "AppStatus.h"
```

```
@interface DetailViewController ()
```

```
@end
```

```
@implementation DetailViewController
```

```
@synthesize naviTitle = _naviTitle;
```

```
-(void)loadView
```

```
{
```

```
    self.view = [[[UIView alloc] initWithFrame:CGRectMake(0, 0, 320, 480)]autorelease];
```

```
    //单例
```

```
    self.title = [AppStatus sharedInstance].contextStr;
```

```
    textField = [[UITextField alloc] initWithFrame:CGRectMake(100, 100, 150, 30)];
```

```
    textField.borderStyle = UITextBorderStyleLine;
```

```
    [self.view addSubview:textField];
```

```
    [textField release];
```

```
    UIBarButtonItem *doneItem =
```

```
    [[[UIBarButtonItem alloc] initWithBarButtonSystemItem:UIBarButtonSystemItemDonetarget:self action:@selector(doneAction:)];
```

```
    self.navigationItem.rightBarButtonItem = doneItem;
```

```
    [doneItem release];
```

```
}
```

```
//这个方法是执行多遍的 相当于刷新view
```

```
-(void)viewWillAppear:(BOOL)animated
```

```
{
```

```
    [super viewWillAppear:animated];
```

```
    tf = (UITextField *)[self.view viewWithTag:1000];
```

```
    tf.text = [AppStatus sharedInstance].contextStr;
```

```
}
```

```
//pop回前一个页面
```

```
-(void)doneAction:(id)sender
```

```
{
```

```
    // 单例传值
```

```
    [AppStatus sharedInstance].contextStr = textField.text;
```

```
    [self.navigationController popToRootViewControllerAnimated:YES];
```

```
}
```

通知传值 谁要监听值的变化，谁就注册通知 特别要注意，通知的接受者必须存在这一先决条件

```
A页面RootViewController.h
```

```
#import <UIKit/UIKit.h>
```

```
#import "DetailViewController.h"
```

```
@interface RootViewController : UIViewController<ChangeDelegate>
```

```
{
```

```
    UITextField *tf;
```

```
}
```

```
@end
```

```
A页面RootViewController.m
```

```
#import "IndexViewController.h"
```

```
#import "DetailViewController.h"
```

```
#import "AppStatus.h"
```

```
@implementation IndexViewController
```

```
{
    [[NSNotificationCenter defaultCenter] addObserver:self
                                           name:@"CHANGE_TITLE" object:nil];
    [super dealloc];
}

-(id)init
{
    if (self = [super init])
    {
        [[NSNotificationCenter defaultCenter] addObserver:self
                                           selector:@selector(change:)
                                           name:@"CHANGE_TITLE"
                                           object:nil];
    }
    return self;
}

-(void)change:(NSNotification *)aNoti
{
    // 通知传值
    NSDictionary *dic = [aNoti userInfo];
    NSString *str = [dic valueForKey:@"Info"];

    UITextField *tf = (UITextField *)[self.view viewWithTag:1000];
    tf.text = str;
}
```

```
-(void)viewWillAppear:(BOOL)animated
{
    [super viewWillAppear:animated];

    /*
    // 单例传值
    UITextField *tf = (UITextField *)[self.view viewWithTag:1000];
    tf.text = [AppStatus sharedInstance].contextStr;
    */
}
@end
```

DetailViewController.h

```
#import <UIKit/UIKit.h>
@protocol ChangeDelegate; //通知编译器有此代理

@interface DetailViewController : UIViewController
{
    UITextField *textField;
}
@end
```

DetailViewController.m

```
#import "DetailViewController.h"
#import "AppStatus.h"

@implementation DetailViewController
@synthesize naviTitle = _naviTitle;

-(void)loadView
{
    UIBarButtonItem *doneItem =
    [[UIBarButtonItem alloc] initWithBarButtonSystemItem:UIBarButtonSystemItemDonetarget:self action:@selector(doneAction:)];
    self.navigationItem.rightBarButtonItem = doneItem;
    [doneItem release];
}

// pop回前一个页面
-(void)doneAction:(id)sender
{
    NSDictionary *dic = [NSDictionary dictionaryWithObject:textField.text forKey:@"Info"];

    [[NSNotificationCenter defaultCenter] postNotificationName:@"CHANGE_TITLE" object:nil userInfo:dic];

    [self.navigationController popViewControllerAnimated:YES];
}
}
```

Block

几种形式的Block

```
//无返回值
void (^block1) (void);
block1 = ^{
    NSLog(@"block demo");
};
block1();

//int返回类型
```

```
{
    int a = 1 ,b =1;
    int c = a+b;
    return c;
};

//有返回 有参数
int (^block3)(int, int)= ^(int a, int b)
{
    int c = a +b;
    return c;
};

NSLog(@"boc=%d",block3(1,2));

//有返回值. 有参数并且可以修改block之外变量的block
static int sum = 10;// _blkic and static关键字 或者 _block int sum = 10
int (^block4) (int) =^(int a)
{
    sum=11;
    int c = sum+a;    //此时sum就是可以修改的了, 若没加static或_block关键字则不能修改block之外变量
    return c;
};

NSLog(@"block4= %d",block4(4));
```

Block传值

例如A(Ablock)页面的值传进B(Bblock)页面

```
在A页面中ABlock.h
@interface Ablock : UIViewController<UITableViewDelegate,UITableViewDataSource>
{
    UITableView *_tableView;
    UILabel *_labe;
    UIImageView *_imagevies;
}
@end

-(void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath
{
    [_tableView deselectRowAtIndexPath:indexPath animated:YES];

    Bblcok *bblock = [[Bblcok alloc] initWithBlock:Block_copy(^(NSString *aBlock){
        labe.text = aBlock;
        NSLog(@"%@",aBlock);

    })];

    bblock.imgviews = imagevies.image;
    bblock.String = labe.text;
    [self.navigationController pushViewController:bblock animated:YES];
    [bblock release];
}
```

```
在A页面中BBlock.h
#import <UIKit/UIKit.h>
typedef void (^MyBlock) (NSString *);
@interface Bblcok : UIViewController
{
    UIImageView *_image;
    UITextField *_aField;
    UIButton *_aButt;
    NSString *_String;
    id _imgviews;
    MyBlock myBlock;
}
@property(n nonatomic,copy)MyBlock myBlock;
@property(n nonatomic,retain) id imgviews;
@property(n nonatomic,retain) NSString *String;
-(id)initWithBlock:(MyBlock)aBlcok;
@end

//
// Bblcok.m
// Blcok
//
// Created by zhu on 13-8-12.
// Copyright (c) 2013年 Zhu Ji Fan. All rights reserved.
//

#import "Bblcok.h"

@interface Bblcok ()

@end

@implementation Bblcok
@synthesize imgviews = _imgviews , String = _String;
@synthesize myBlock = _myBlock;
-(id)initWithBlock:(MyBlock)aBlcok
{
    if (self = [super init])
    {
        self.myBlock = aBlcok;
    }
}
```

```
-(void) dealloc
{
    [super dealloc];
}

-(void) loadView
{
    UIControl *cont = [[UIControl alloc] initWithFrame:CGRectMake(0, 0, 320, 568-44)];
    [cont addTarget:self action:@selector(Clcik) forControlEvents:UIControlEventTouchUpInside];
    self.view = cont;

    aField = [[UITextField alloc] initWithFrame:CGRectMake(60, 10, 160, 30)];
    aField.borderStyle = UITextBorderStyleLine;
    aField.placeholder = self.String;
    [self.view addSubview:aField];

    aButt = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    aButt.frame = CGRectMake(60, 50, 70, 30);
    [aButt setTitle:@"修改" forState:0];
    [aButt addTarget:self action:@selector(aButtClcik:) forControlEvents:UIControlEventTouchUpInside];
    [self.view addSubview:aButt];

    image = [[UIImageView alloc] initWithFrame:CGRectMake(60, 100, 210, 260)];
    image.backgroundColor = [UIColor blueColor];
    image.image = self.imgviews;
    [self.view addSubview:image];
    [image release];
}

-(IBAction)aButtClcik:(id)sender
{
    NSString *sting = aField.text;
    myBlock(sting);
    [self.navigationController popToRootViewControllerAnimated:YES];
}

-(void)Clcik
{
    [aField resignFirstResponder];
}

-(void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.
}

-(void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

@end
```

来自: [ccocshq](#) > [《其他》](#)
上一篇: [UIColor RGB颜色对照表](#)
下一篇: [如何使用XCode创建ipa文件及上传应用程序](#)

ios视频教程 学习数据库 苹果app开发 ios培训 sql数据库学习

ios开发工程师 数据库入门

sql语句学习 手机app开发 数据库学习视频 数据库学习 上海程序员培训

转藏到我的图书馆 献花(0) 分享到微信 以文找文 分享:

类似文章 更多 热门推荐

ios多线程开发 NSThread

IOS UI 第四篇: 基本UI

Objective-C Beginner's Guide

stupid教程

Objective-C中一种消息处理方法performS...

iphone开发常用代码

苹果开发重要语法知识

socket第三方库 AsyncSocket(GCDAsyncS...

1 TOP:白富美是这样做月子的

2 赞:我20年尿糖病终于好了!

3 通过号码监控,了解对方在干嘛?

4 租叉车吧!且租且珍惜哦!

5 上海强力磁铁批发商

6 油烟管道清洗 全市上门服务

7 暖通设备 找集家网 疯狂折扣

8 如何用镜头端出满席宴会?

猜你喜欢

换一组

360doc 个人图书馆

首 页

阅览室

馆友

我的图书馆



十句经典话，写尽中



慧人、慧语、慧悟



为人处世105句



婚姻——一个月，30万



社会生存法则



防水运动蓝牙耳机
4.0双耳入耳无线
¥ 399.00



迷你蓝牙耳机通用
手机电脑上班视频
¥ 128.00



美国PLUFY运动好
音质运动蓝牙4.0
¥ 338.00

发表评论：

您好，请 [登录](#) 或者 [注册](#) 后再进行评论

其它帐号登录：