在项目中安装GHUnit框架

新建Window-based Application项目，命名为GHUnitTest。

使用Add->New Target…添加一个target，使用模板 iPhone OS-> Cocoa Touch->Application，target 命名为tests（或者别的什么）。

把下载到的GHUnitIOS.framework文件拷贝到项目目录下。在frameworks中选择GHUnitIOS.framework，打开info窗口，切换到Targets面板，确保已正确地包含在了tests这个target中：



选择Targets下的tests，打开info窗口，在General页，通过左下角的+号按钮，把以下框架也包含进LinkedLibraries中：

CoreGraphics.framework

Foundation.framework

UIKit.framework

在Build面板，确保Framework Search Paths中已包含了GHUnitIOS.framework所在的路径；在OtherLinker Flags中加入-ObjC和-all\_load。

将Tests-Info.plist文件中Main nib file base name一行删除。

将GHUnitIOSTestMain.m文件加入到项目中，下载地址：<http://github.com/gabriel/gh-unit/blob/master/Project-iOS/GHUnitIOSTestMain.m>。

添加时，注意确保将文件包含到tests中（注意Add To Targets栏）：



在Other Sources中新建一个预编译头文件tests\_Prefix.pch，并在其中加入一行：#import<GHUnitIOS/GHUnit.h>。然后在tests的Build设置中将Prefix Header设置为tests\_Prefix.pch，这样就不需要每个测试类都import了。

3、创建测试类

新建Objective C类MyTest。修改MyTest.h，将父类由NSObject修改为GHTestCase。修改MyTest.m，实现测试方法（方法名以test开头）：

- (void)testStrings {

    NSString \*string1= @"a string";

   GHTestLog(@"I can log to the GHUnit test console: %@",string1);

    // Assert string1is not NULL, with no custom error description

   GHAssertNotNULL(string1, nil);

    // Assert equalobjects, add custom error description

    NSString \*string2= @"a string";

    GHAssertEqualObjects(string1,string2, @"A custom error message. string1 should be equal to: %@.",string2);

}

把当前Build Configure设置为Simulator|Debug|tests：

[](http://static.oschina.net/uploads/img/201201/09180141_NgSJ.png%22%20%5Ct%20%22_blank)

点击“Build and Run”，弹出模拟器窗口，点击右上角的Run，测试运行结果如下：



测试结果在界面和控制台中都有显示。