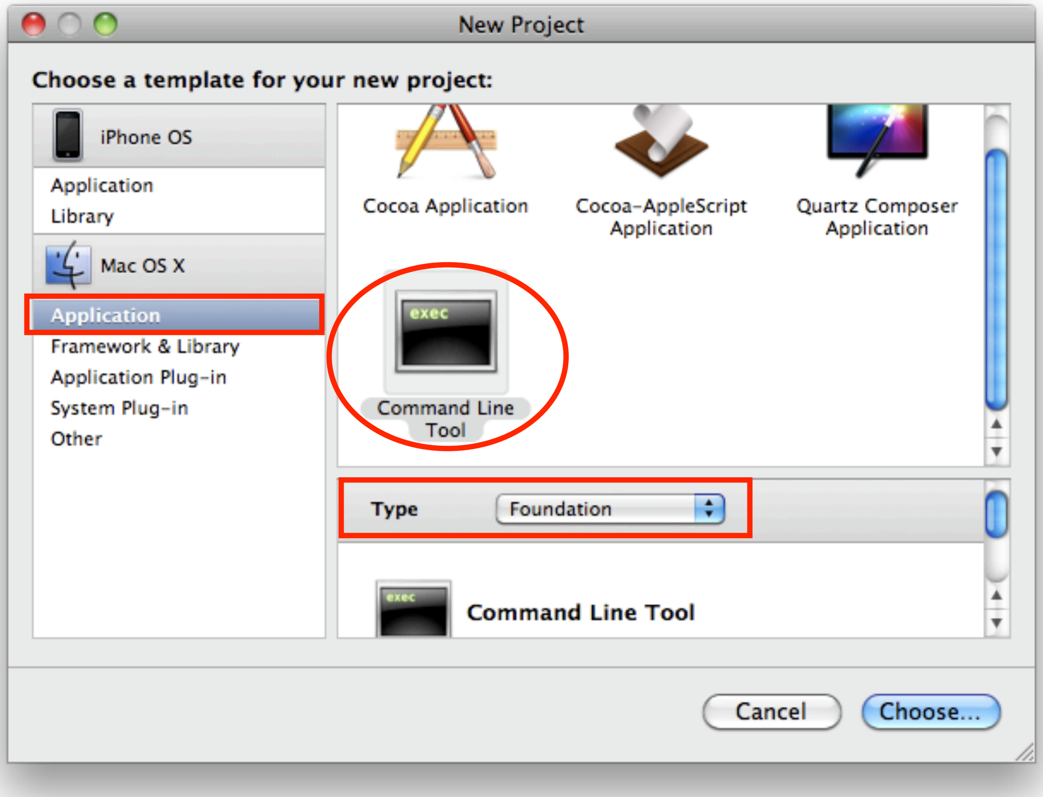
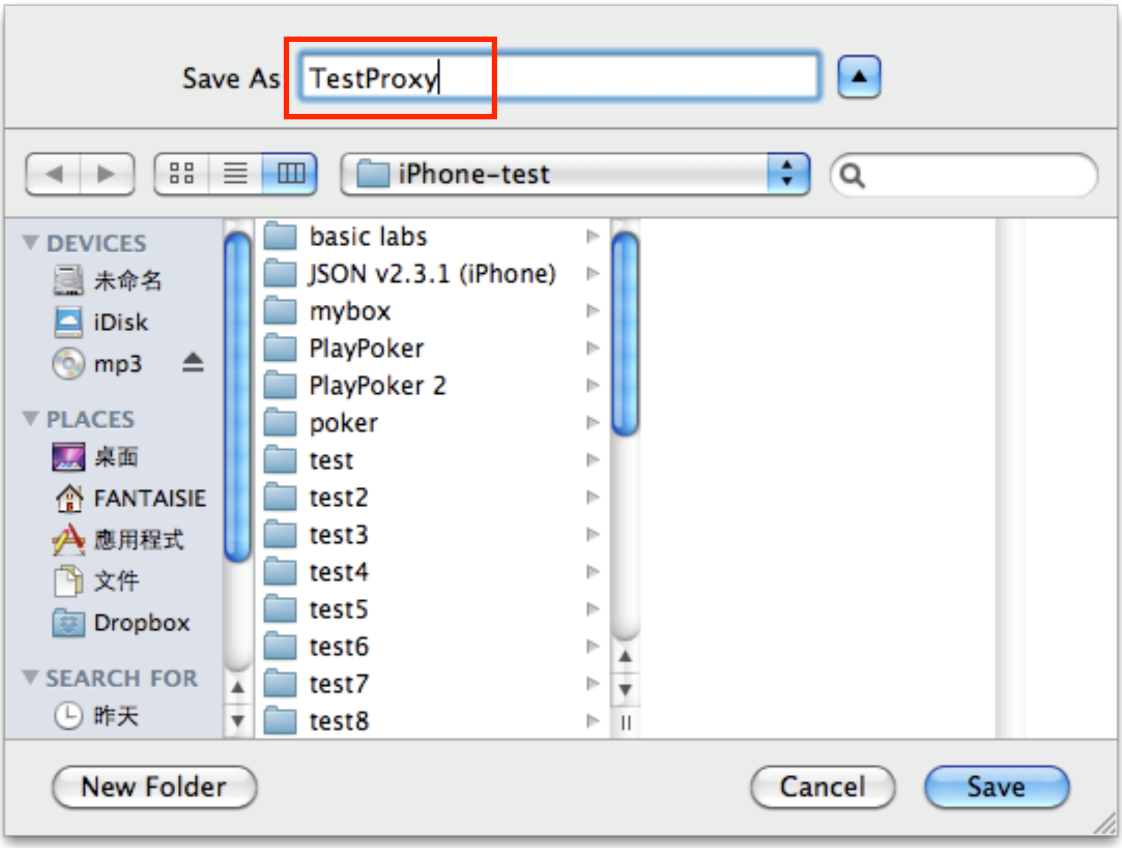
Lab TestProxy

Step1. 在File開啓一個新的project, 選擇 MAC OS X的Command line Tool, Type選擇 Foundation, 將project命名為 TestProxy

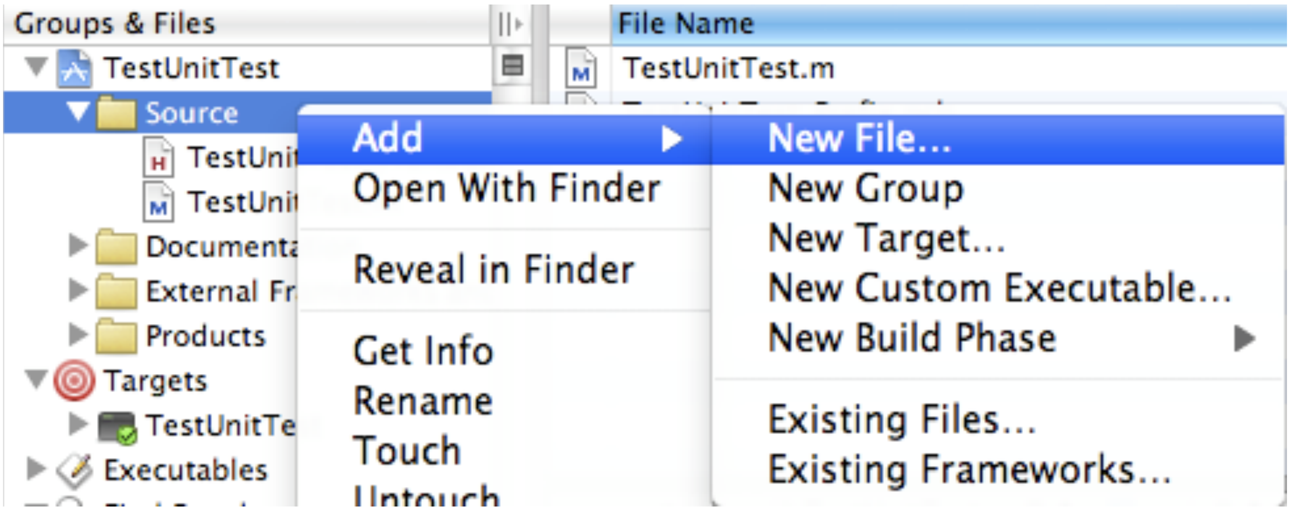


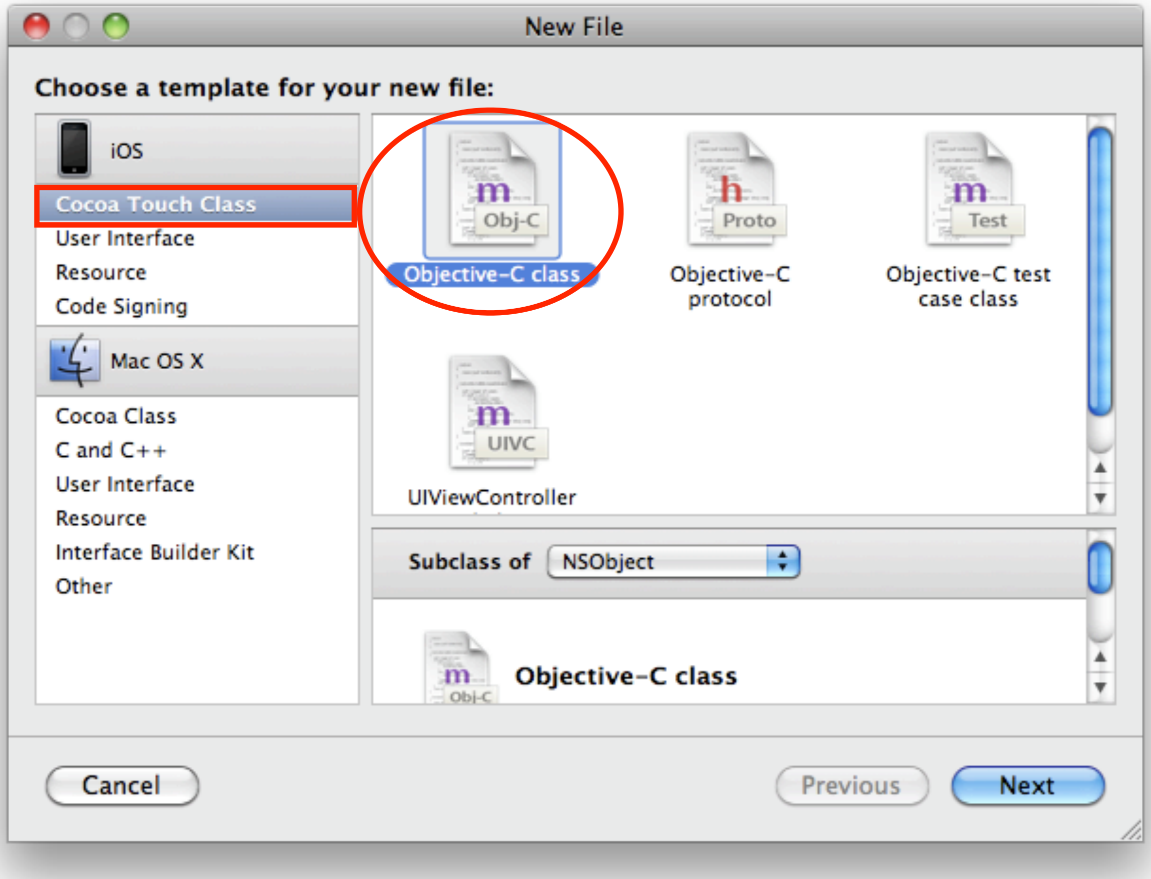
Type: Foundation



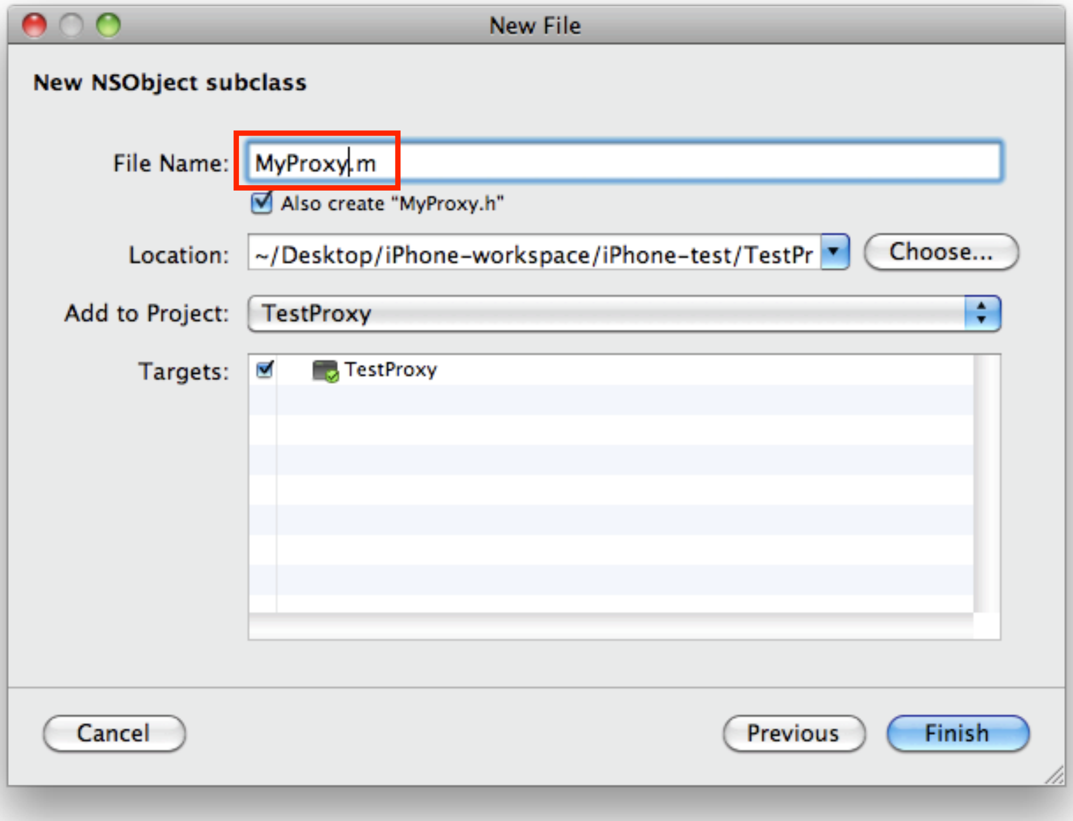
Save As: TestProxy

Step2. 在Xcode左邊Groups & Files 視窗中, 在Source這個路徑下點右鍵(若無滑鼠ctrl+點擊)選擇Add > New File...來增加新的Objective-C class





Step3. 將這個class命名為MyProxy.m



New NSObject subclass

File Name: MyProxy.m

☑ Also create "MyProxy.h"

Location: ~/Desktop/iPhone-workspace/iPhone-test/TestPr

Add to Project: TestProxy

Targets: ☑ TestProxy

Step4. 在MyProxy.h裡, 先將原來繼承的 NSObject 改成 NSProxy, 宣告我們會代理的Object (因無法知道為什麼Class, 設為id), 並設定property來宣告accessor.

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

@interface MyProxy : NSProxy {
    id targetObject;
}

@property (retain, nonatomic) id targetObject;

@end
```

Step5. 在MyProxy.m裡, 除了實作建構子 init 之外(必需), 還必須實作覆寫兩個 NSProxy 的 method: `forwardInvocation: 和 methodSignatureForSelector:`

```
#import "MyProxy.h"


@implementation MyProxy
@synthesize targetObject;

-(id) init{
    return self;
}

-(void)forwardInvocation:(NSInvocation *)anInvocation {
    [anInvocation setTarget:targetObject];
    [anInvocation invoke];
}

-(NSMethodSignature *)methodSignatureForSelector:(SEL)aSelector {
    return [targetObject methodSignatureForSelector:aSelector];
}

@end
```

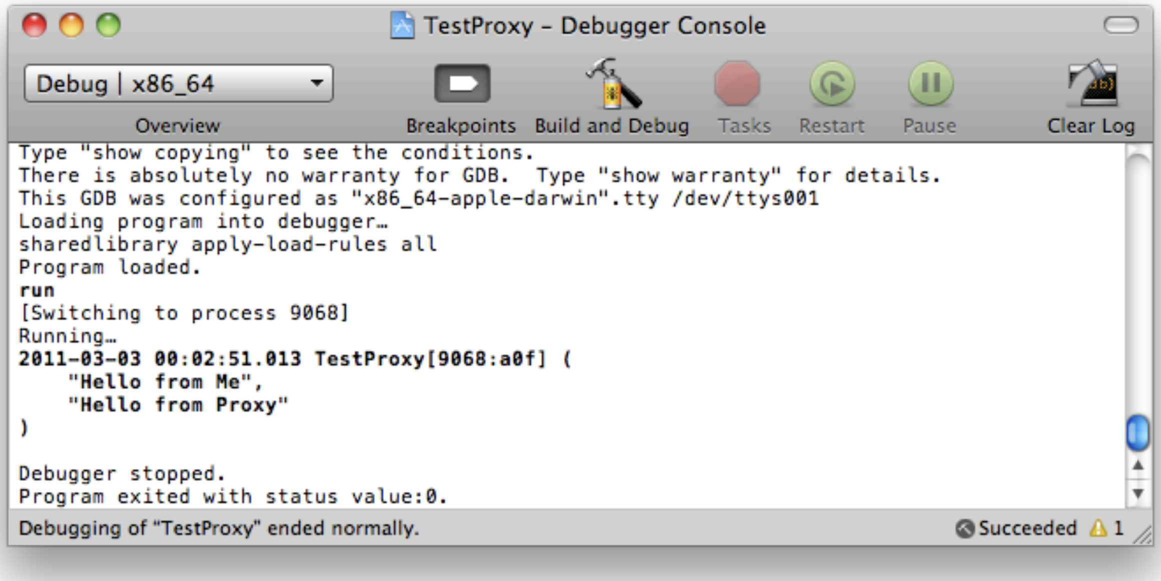
Step6. 在TestProxy.m裡，先 `import "MyProxy.h"` ，然後將印出Hello, World!的這行去掉, 新增一個NSMutableArray的物件newArray, 以及一個MyProxy的物件newProxy, 在Array放進一個NSString的物件之後,接下來就把工作交由newProxy代理,然後由newProxy來新增另外一個NSString的物件,最後透過NSLog列印出來.

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

int main (int argc, const char * argv[]) {
    NSAutoreleasePool * pool = [[NSAutoreleasePool alloc] init];

    // insert code here...
    //NSLog(@"Hello, World!");
    NSMutableArray *newArray = [NSMutableArray arrayWithObject:@"Hello
from Me"];
    MyProxy *newProxy = [[MyProxy alloc] init];
    newProxy.targetObject = newArray;
    [newProxy addObject:@"Hello from Proxy"];
    NSLog(@"%@", newProxy.targetObject);
    [pool drain];
    return 0;
}
```

Step7. Build and Run (Command + enter)
在Xcode主頁上按下Build and Run, 或是在Build > Build and Run
印出由newArray自己放進的NSString和由newProxy新增的一個NSString

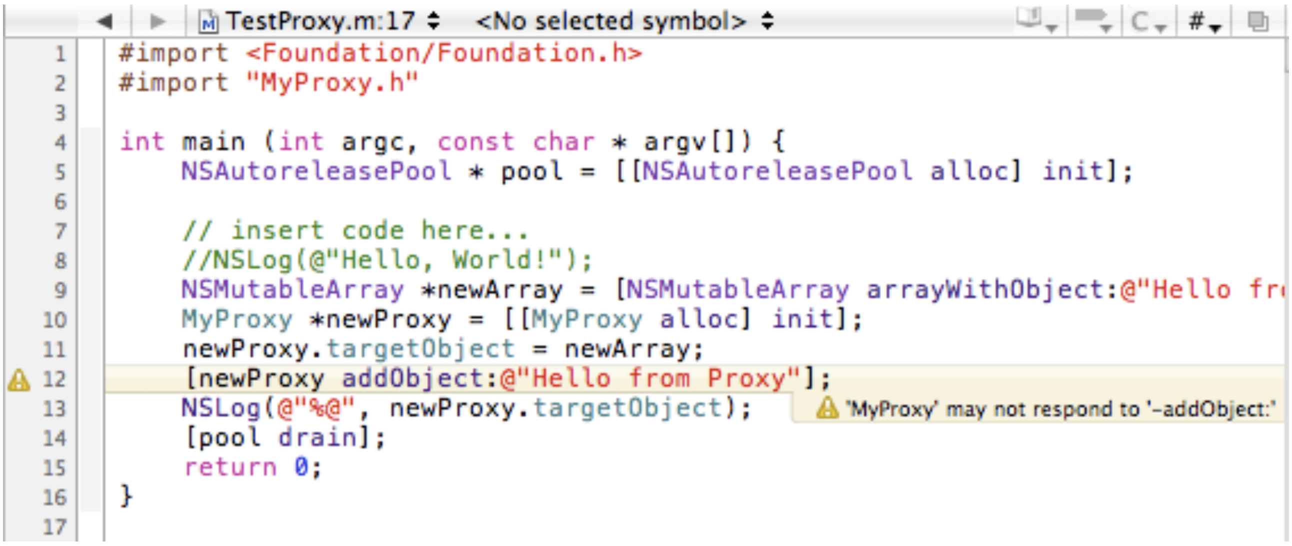


```
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "x86_64-apple-darwin".tty /dev/ttys001
Loading program into debugger…
sharedlibrary apply-load-rules all
Program loaded.
run
[Switching to process 9068]
Running…
2011-03-03 00:02:51.013 TestProxy[9068:a0f] (
    "Hello from Me",
    "Hello from Proxy"
)

Debugger stopped.
Program exited with status value:0.
```

Debugging of "TestProxy" ended normally.

但在 TestProxy.m 會顯示 **"MyProxy" may not respond to "-addObject:"**



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

int main (int argc, const char * argv[]) {
    NSAutoreleasePool * pool = [[NSAutoreleasePool alloc] init];

    // insert code here...
    //NSLog(@"Hello, World!");
    NSMutableArray *newArray = [NSMutableArray arrayWithObject:@"Hello fr
    MyProxy *newProxy = [[MyProxy alloc] init];
    newProxy.targetObject = newArray;
    [newProxy addObject:@"Hello from Proxy"];
    NSLog(@"%@", newProxy.targetObject);
    [pool drain];
    return 0;
}
```

⚠ 'MyProxy' may not respond to '-addObject:'