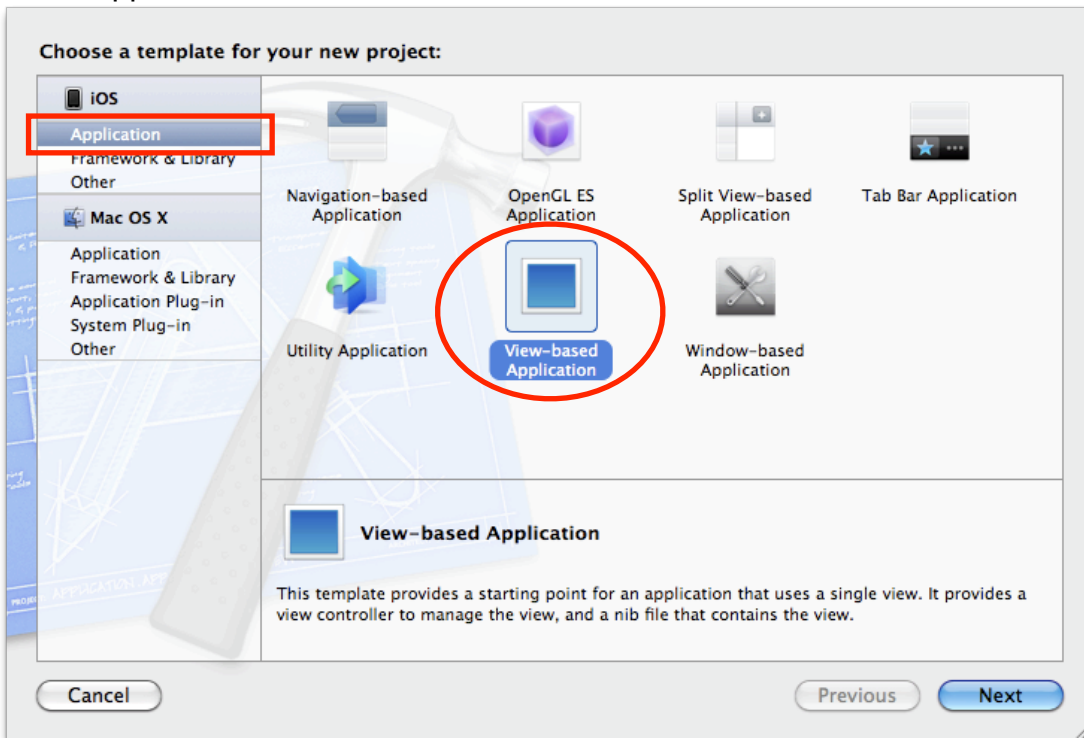


# Lab XMLAndPredicate

Step 1. 在File>New>New Project開啓一個新的專案, 在iOS的Application目錄裡面選擇 view based application

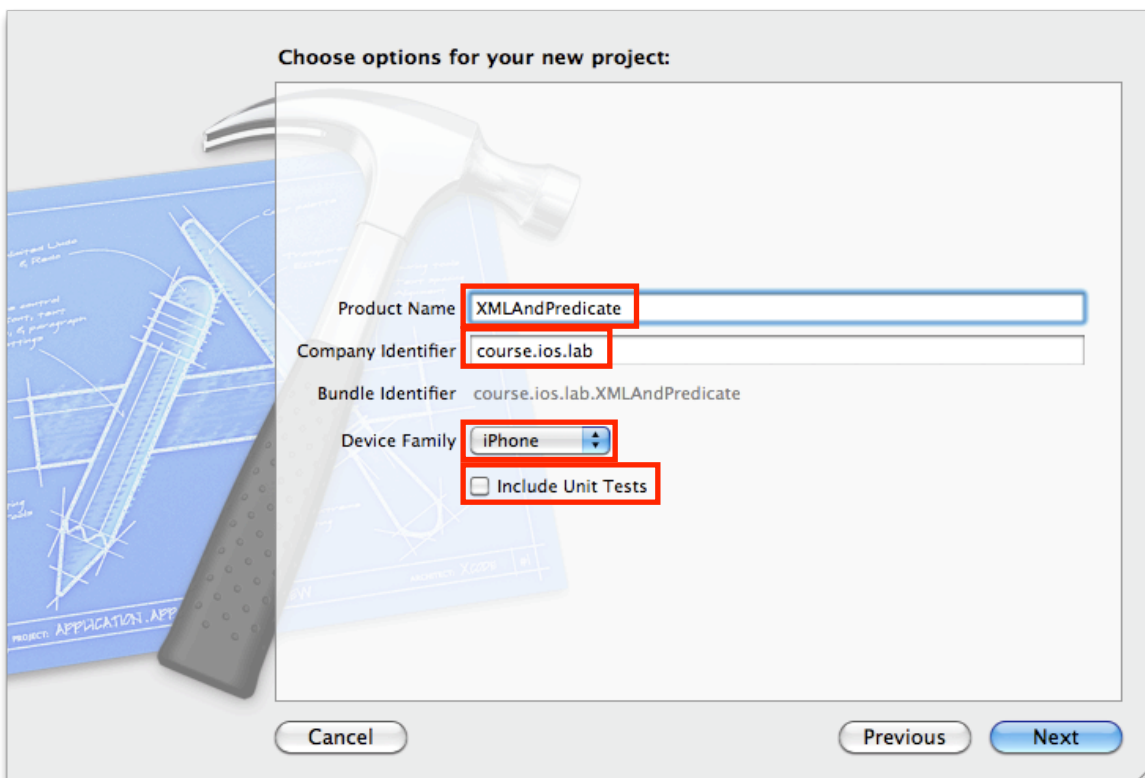


Step 2. 並將此專案命名為 **XMLAndPredicate**

Company Identifier是填入Bundle的名稱,在此統一填入**course.ios.lab** (也可自行填入)

Device Family選擇**iPhone**

Include Unit Tests是做語意邏輯測試用,可勾選也可不勾選,在此我們統一不勾選, 存檔



Step 3. 到 [http://www.tbxml.co.uk/TBXML/TBXML\\_Free.html](http://www.tbxml.co.uk/TBXML/TBXML_Free.html) 去下載TBXML這個XML Parser, 或是直接在Google搜尋TBXML, 在右邊Download選擇 **TBXML.zip** 來下載

The screenshot shows the 'TBXML - Free' website. The main content area has sections for 'About', 'Performance', and 'Design Goals'. A 'Download (Version 1.4)' section lists 'TBXML.zip' and 'TBXML-Books.zip'. A sidebar on the right contains 'Last Updated' (20th May 2010), 'Download (Version 1.4)' (with 'TBXML.zip' highlighted by a red box), 'Download (for Java)' (listing 'TBXML-java.zip' and 'Written by Julien Foltz'), and a 'Donate' section with a 'Donate' button and logos for Mastercard, American Express, Visa, and others.

## TBXML - Free

TBXML Free   API   Guides   Blog   Forum   Contact

### About

TBXML is a light-weight XML document parser written in Objective-C designed for use on Apple iPad, iPhone & iPod Touch devices. TBXML aims to provide the fastest possible XML parsing whilst utilising the fewest resources. This requirement for absolute efficiency is achieved at the expense of XML validation and modification. It is not possible to modify and generate valid XML from a TBXML object and no validation is performed whatsoever whilst importing and parsing an XML document.

### Performance

Check out this post for a good comparison of XML parsers. "How To Chose The Best XML Parser for Your iPhone Project"

### Design Goals

The design goals for TBXML are:

- ✱ XML files conforming to the W3C XML spec 1.0 should be passable
- ✱ XML parsing should incur the fewest possible resources
- ✱ XML parsing should be achieved in the shortest possible time
- ✱ It shall be easy to write programs that utilise TBXML

### Download (Version 1.4)

TBXML.zip  
TBXML-Books.zip

**Last Updated**  
20th May 2010

**Download (Version 1.4)**  
**TBXML.zip**  
TBXML-Books.zip





**Download (for Java)**  
TBXML-java.zip  
Written by Julien Foltz

**Donate**  
If you find TBXML useful and would like to donate, please do so via the paypal link below. All donations are very much appreciated and will help me continue to support and enhance the TBXML of

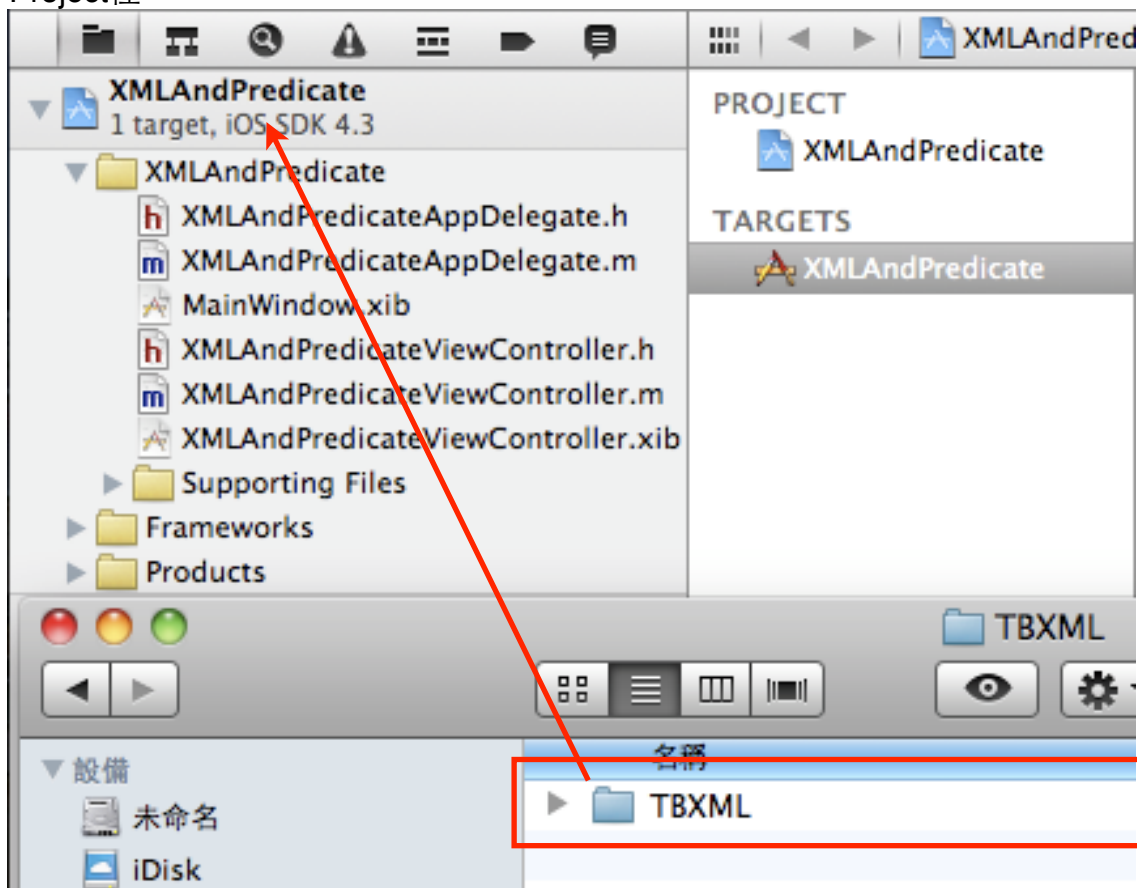
**Donate**

MasterCard American Express VISA VISA

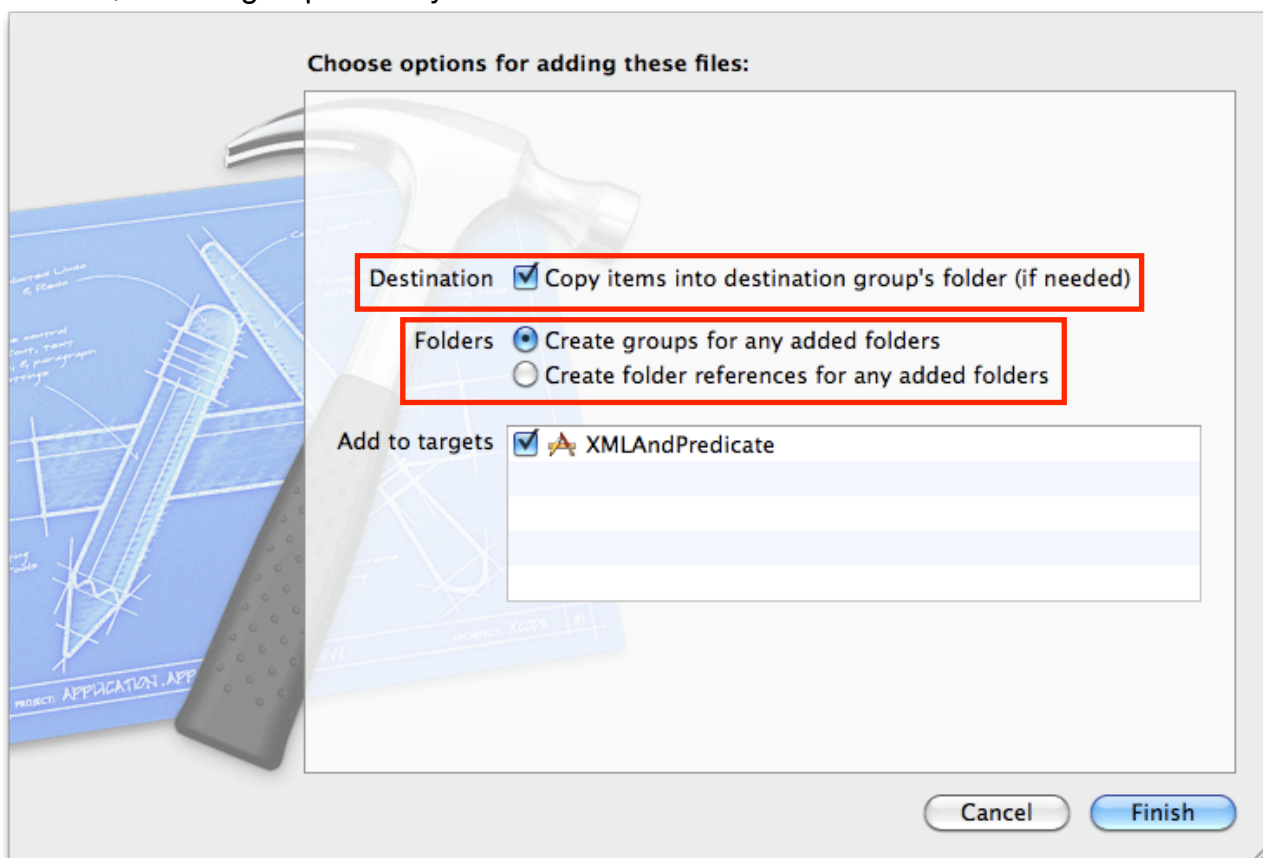
Step 4. 下載完後解壓縮 **TBXML.zip** 裡面分別有四個檔案

名稱
 NSDataAdditions.h
 NSDataAdditions.m
 TBXML.h
 TBXML.m

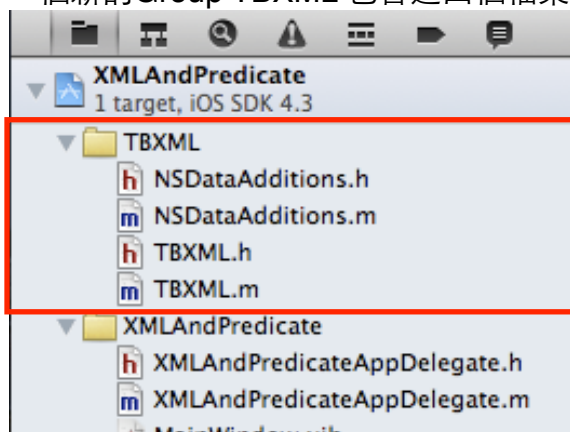
Step 5. 將包含四個檔案的資料夾TBXML整個拖拉到剛剛新增的 XMLAndPredicate 這個 Project 裡



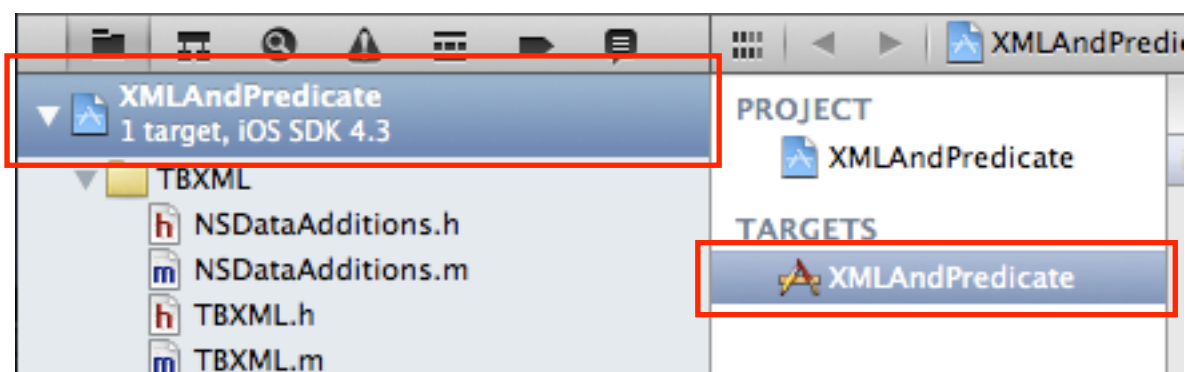
勾選 Copy items into destination group's folder (if needed)  
以及選擇 Create groups for any added folders



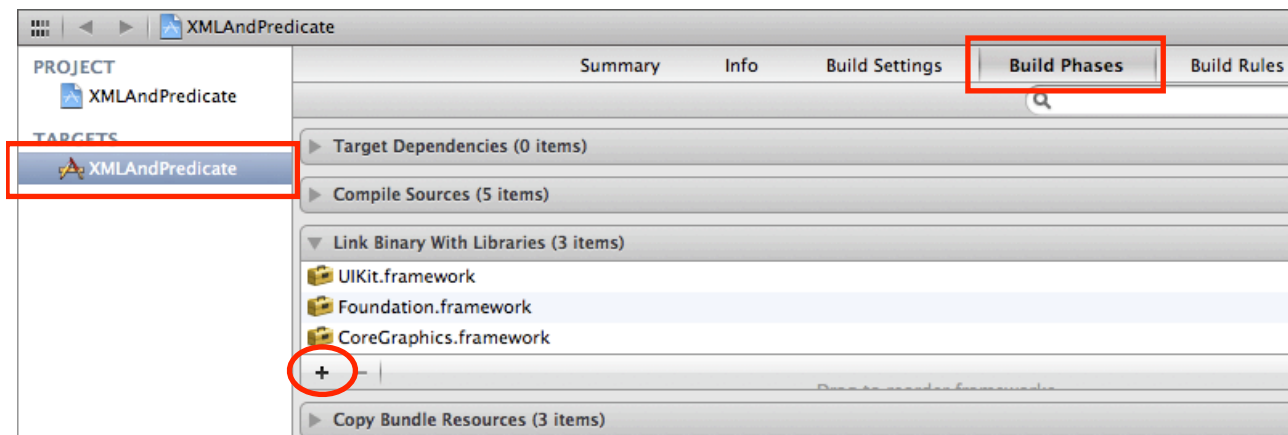
一個新的Group TBXML 包含這四個檔案Copy進我們的Project裡了



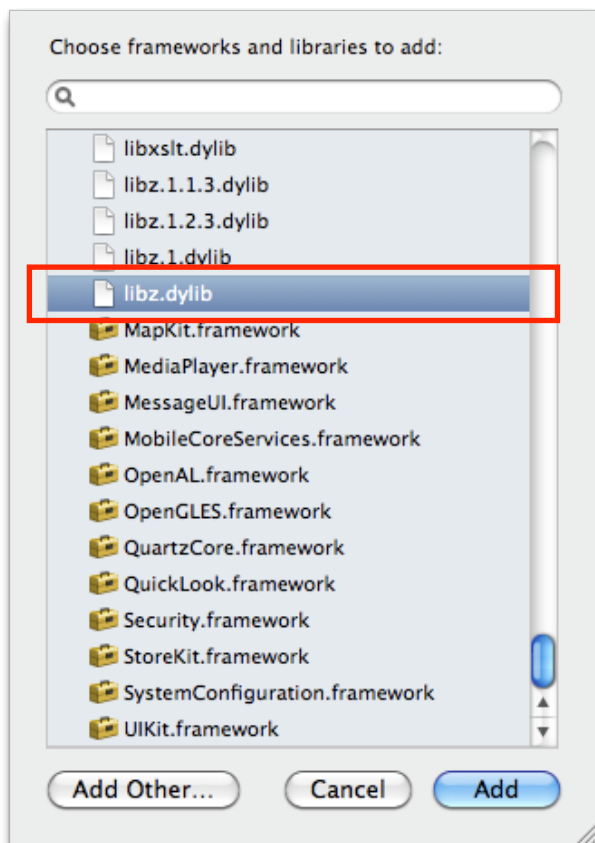
Step 6. 點選我們的Project XMLAndPredicate, TARGETS選擇 XMLAndPredicate



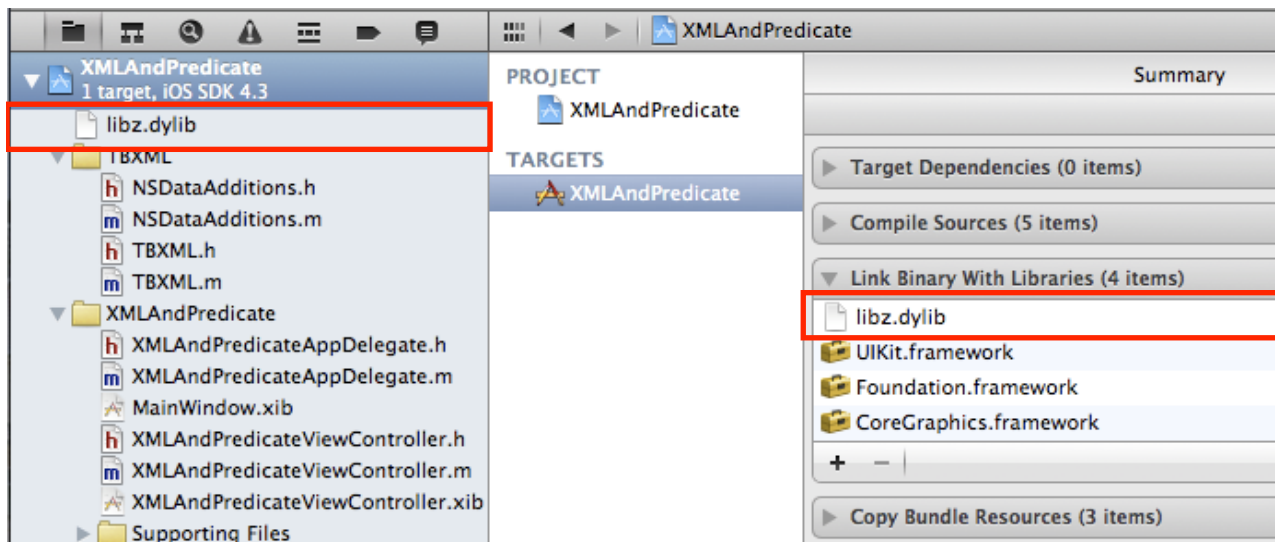
在Build Phases這個Tab裡面的Link Binary With Libraries這欄點選+號加入新的Library



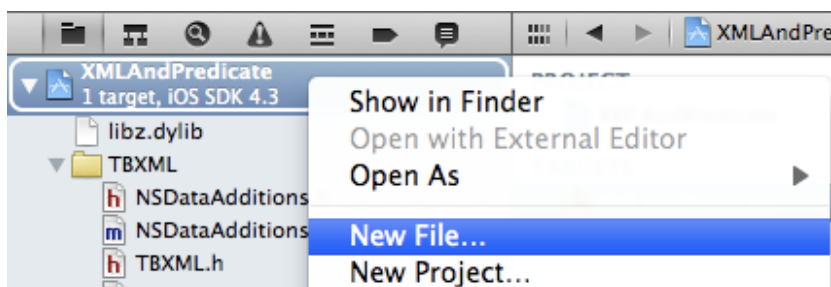
選擇 libz.dylib 這個Library, 然後Add



則 libz.dylib出現在Link Binary With Libraries, 也出現在我們Project中成為一個Reference

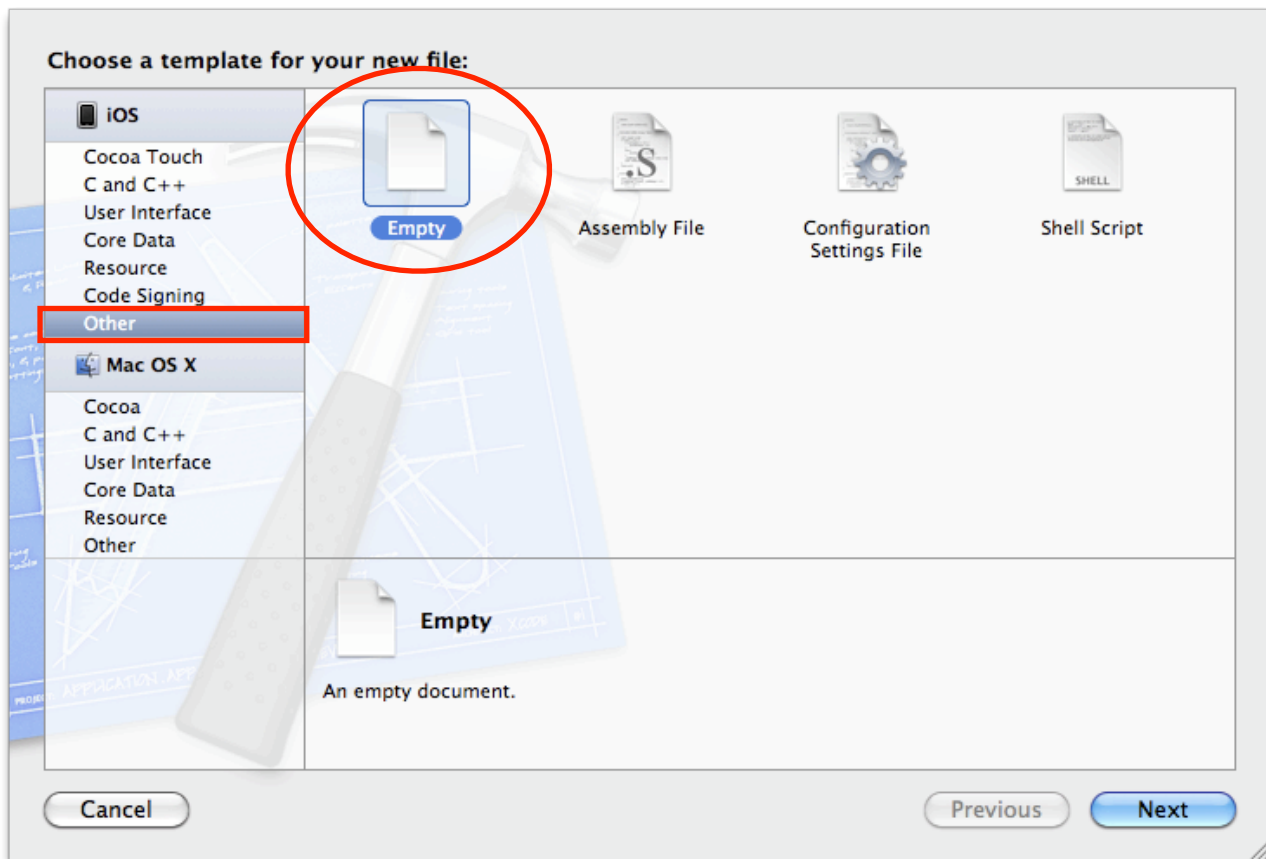


Step 7. 接下來New一個新的File

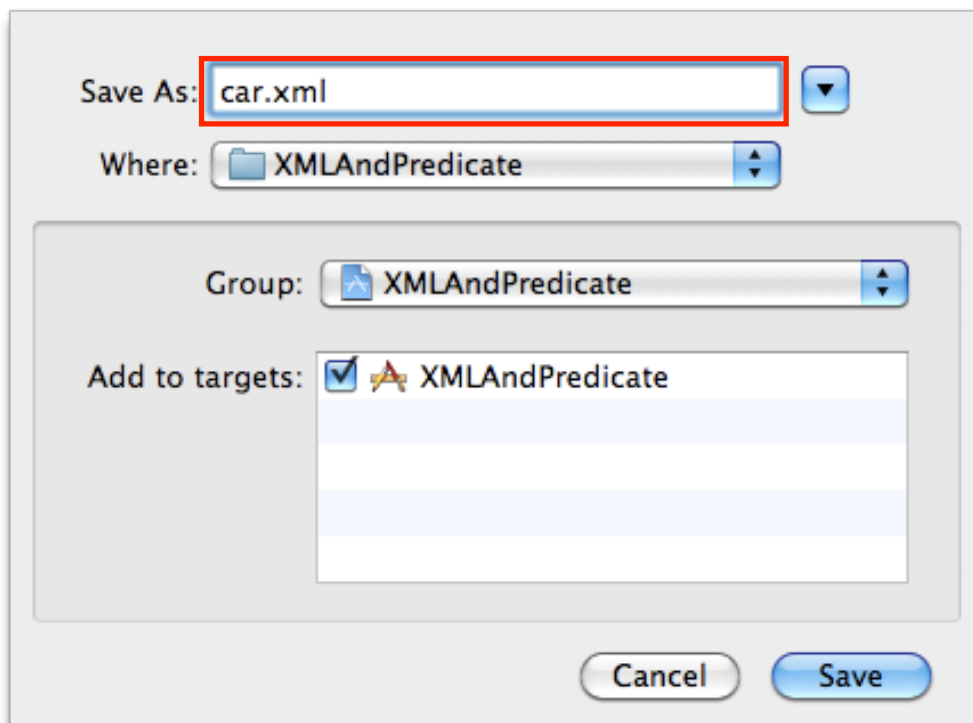




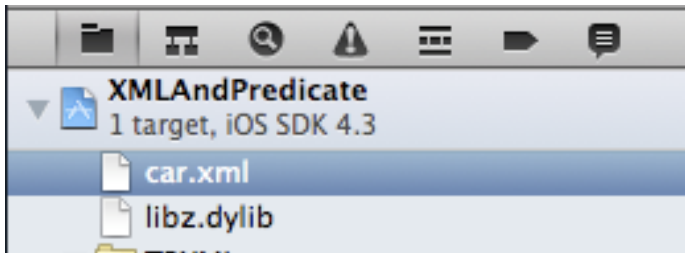
選擇 Other 裡面的 Empty



命名為 car.xml



就產生一個新的XML檔在我們的Project中



Step 8. 開啓car.xml開始如下撰寫XML檔內的內容

```
<car>
  <item>
    <price> 40 </price>
    <year> 1960 </year>
  </item>
  <item>
    <price> 60 </price>
    <year> 1930 </year>
  </item>
  <item>
    <price> 80 </price>
    <year> 1980 </year>
  </item>
</car>
```

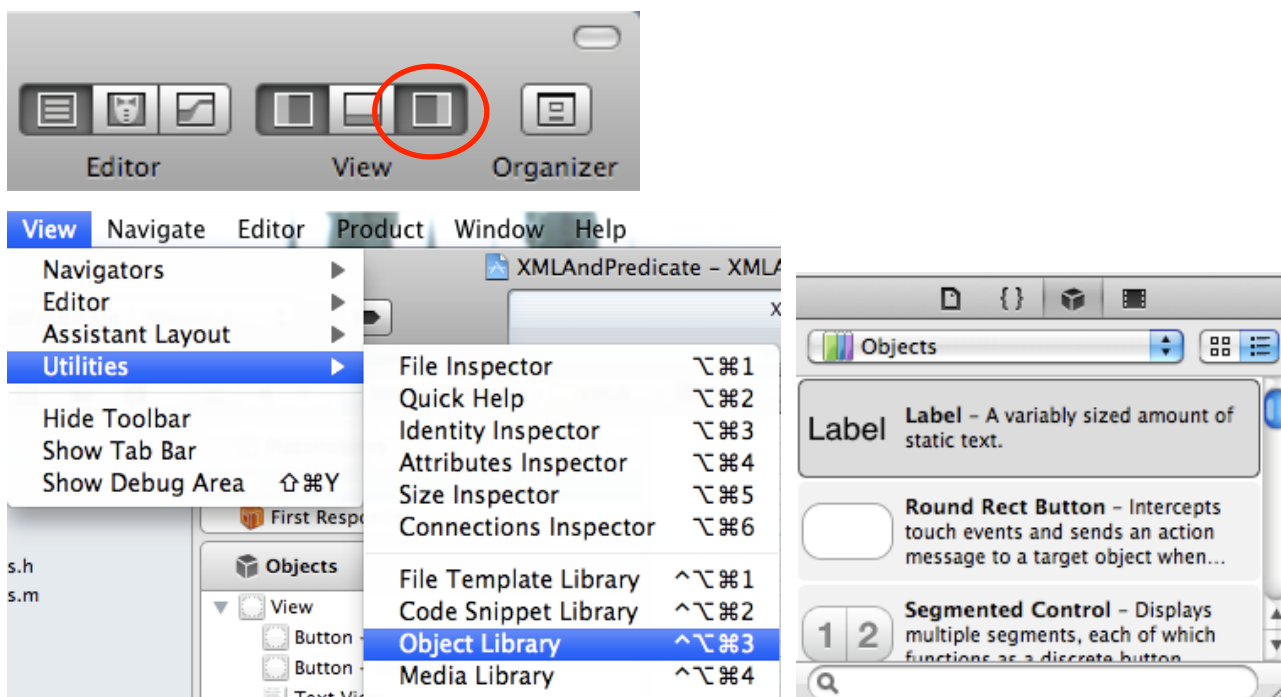
Step 9. 開啓 XMLAndPredicateViewController.h, 加入三個IBOutlet元件,包括兩個 UITextField和一個UITextView, 再加入一個NSMutableArray來放置Parse出來的資訊, 最後再加入兩個IBAction去抓取想要的資訊

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

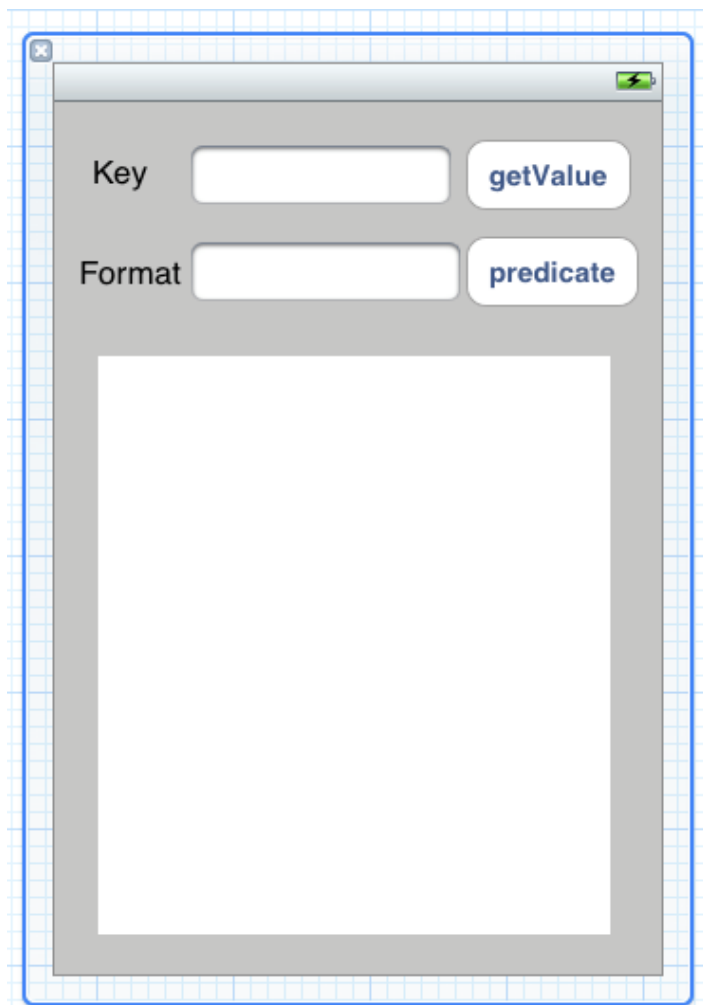
@interface XMLAndPredicateViewController : UIViewController {
    IBOutlet UITextField *inputKey;
    IBOutlet UITextField *inputPredicate;
    IBOutlet UITextView *resultView;

    NSMutableArray * arrayWithDict;
}
- (IBAction)getValue;
- (IBAction)predicateResult;
@end
```

Step 10. 開啓XMLAndPredicateViewController.xib, 在Xcode視窗右上角選擇顯示視窗點下View右邊按鈕,或是View > Utilities 裡選擇任意選擇一個選項即會打開這個視窗分頁,在此選擇Object Library來方便選擇和新增UI元件

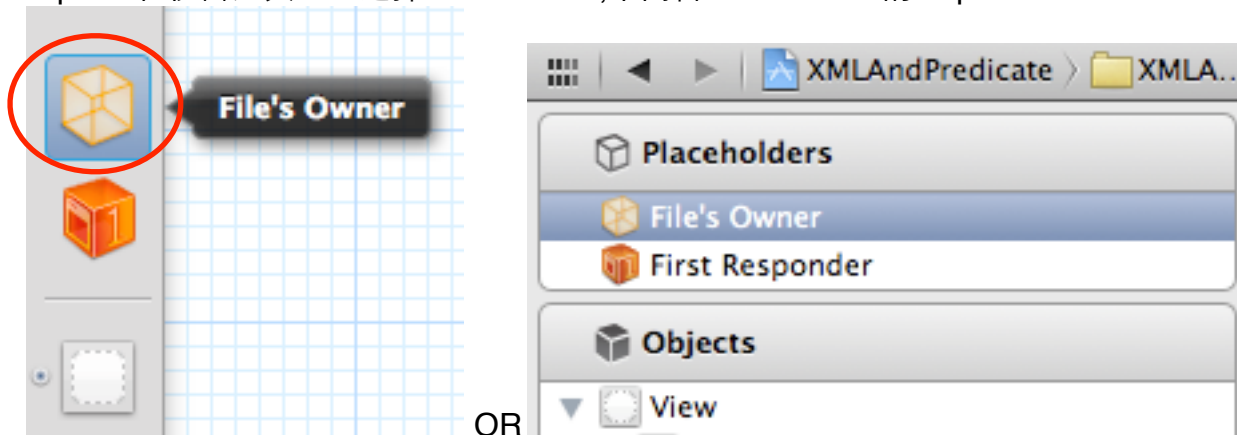


如下在View上加入兩個Label作為Title分別叫做Key和Format, 再加入兩個TextField, 兩個Button, 以及在下方加入一個TextView來顯示出想要的資訊,把TextView原先預設的內容刪掉

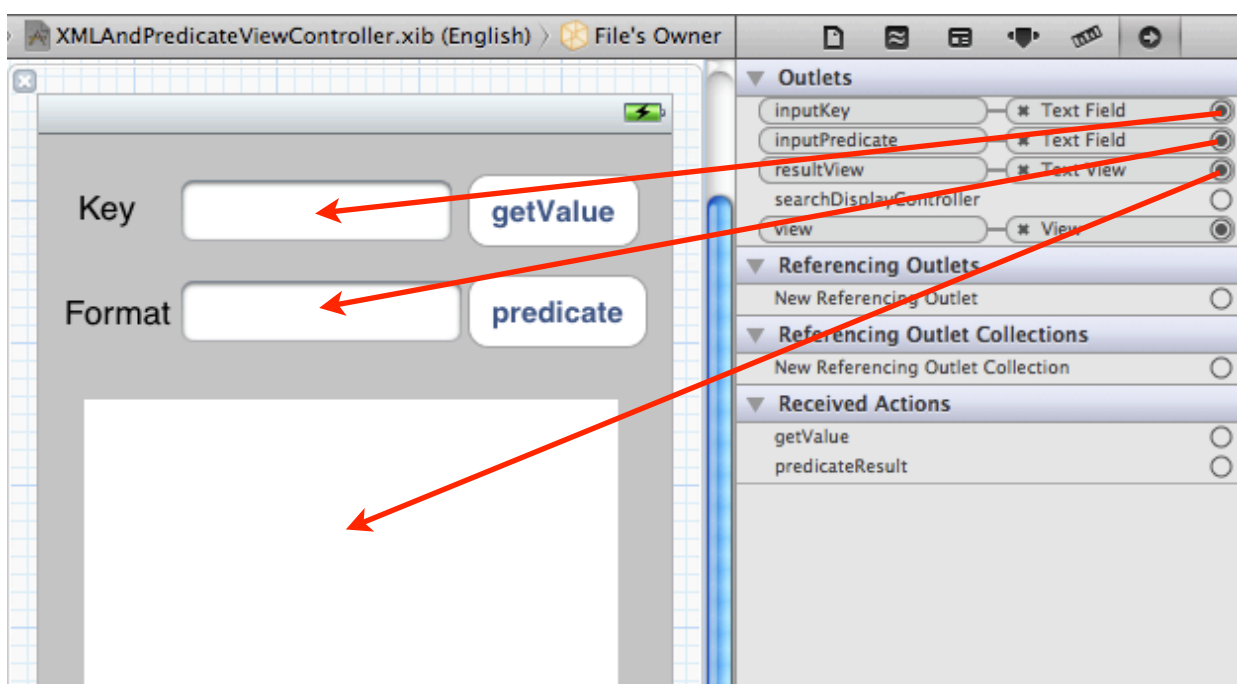




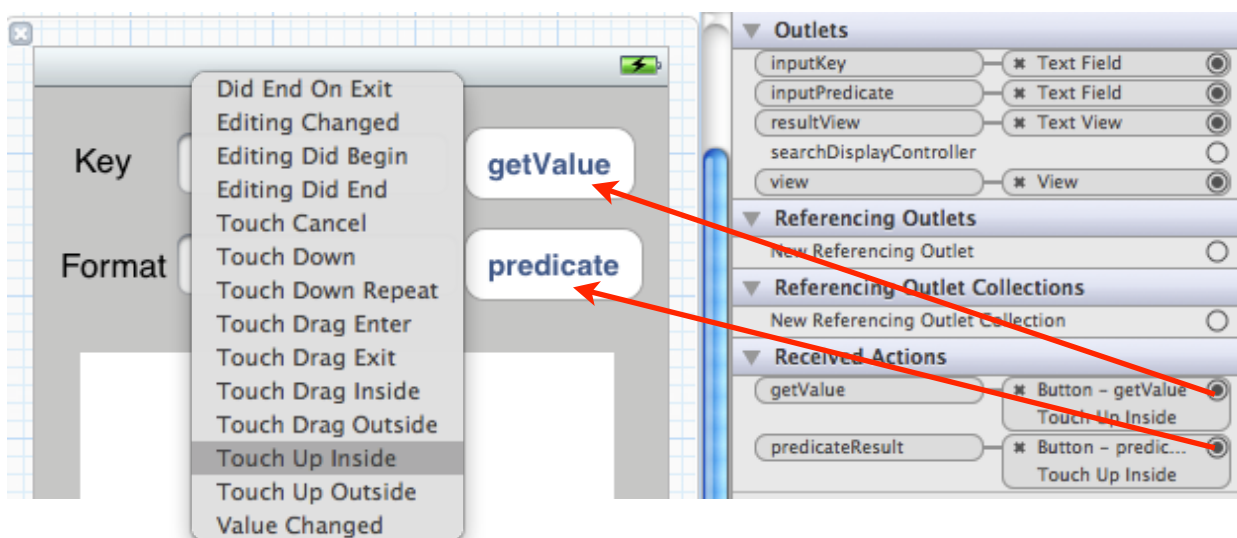
Step 11. 在視窗分頁左邊選擇File's Owner,來開啓File's Owner的Inspector



Step 12. 將三個IBOutlet的元件和View上的元件作連結



Step 13. 將兩個IBAction和View上的兩個Button的Touch Up Inside作連結



Step 14. 開啟 XMLAndPredicateViewController.m , 首先先 `#import "TBXML.h"` , 然後找到 `viewDidLoad` 將 `Mark` 去掉, 加入下面的程式, 去將 `car.xml` 裡面的資料 `parse` 出來, 先放進一個 `NSMutableDictionary` 裡面, 再放進一個 `NSMutableArray` 裡面儲存

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    arrayWithDict = [[NSMutableArray alloc] init];
    TBXML * xmlDoc = [TBXML tbxmlWithXMLFile:@"car.xml"];
    TBXMLElement * root = xmlDoc.rootXMLElement;
    TBXMLElement * child = root->firstChild;

    while (child) {
        TBXMLElement * son = child->firstChild;
        NSMutableDictionary * item = [NSMutableDictionary dictionary];
        while (son) {
            [item setObject:[NSNumber numberWithInt:[TBXML
textForElement:son] intValue]] forKey:[TBXML elementName:son]];
            son = son->nextSibling;
        }
        [arrayWithDict addObject:item];
        child = child->nextSibling;
    }
}
```

Step 15. 同樣在 XMLAndPredicateViewController.m , 找到 `-(void)dealloc{}` , 加入下面程式, 讓在 `dealloc` 時把一些我們新增的元件 `release` 掉

```
- (void)dealloc
{
    [inputKey release];
    [inputPredicate release];
    [resultView release];
    [arrayWithDict release];
    [super dealloc];
}
```

Step 16. 同樣在 XMLAndPredicateViewController.m , 實作我們加入的兩個 `IBAction`

```
- (IBAction)getValue{
    if ([inputKey.text isEqualToString:@""]) {
        resultView.text = [arrayWithDict description];
    }else {
        NSString * theKey = inputKey.text;
        NSArray * values = [arrayWithDict valueForKey:theKey];
        NSString * resultString = [NSString stringWithFormat:@"%s\n%@",
theKey, values];
        resultView.text = resultString;
    }
    [inputKey endEditing:YES];
}
```

```

- (IBAction)predicateResult{
    if ([inputPredicate.text isEqualToString:@""]) {
        resultView.text = [arrayWithDict description];
    }else {

        NSString * inputFormat = inputPredicate.text;
        NSPredicate * myFavor = [NSPredicate
predicateWithFormat:inputFormat];

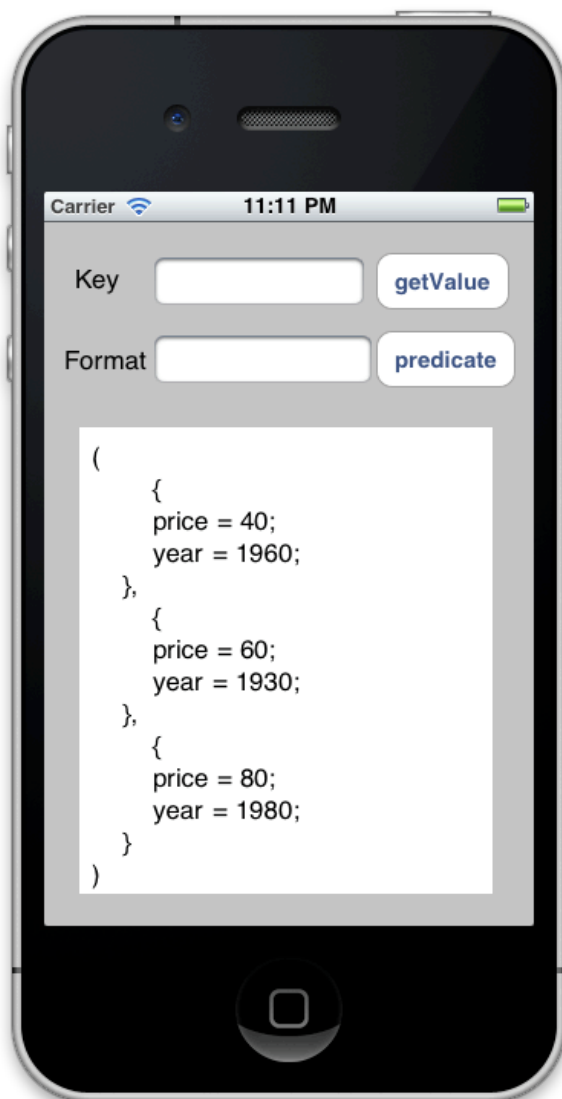
        NSArray * values = [arrayWithDict
filteredArrayUsingPredicate:myFavor];
        NSString * resultString = [NSString stringWithFormat:@"%@\n%@",
inputFormat, values];
        resultView.text = resultString;

    }
    [inputPredicate endEditing:YES];
}

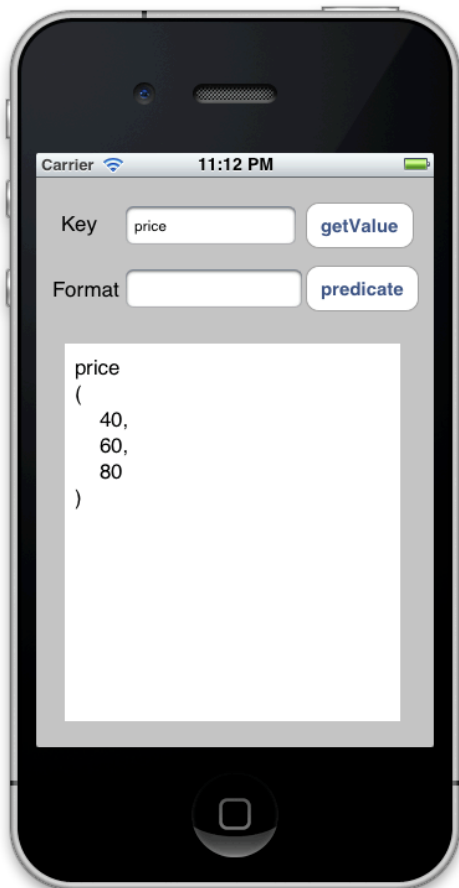
```

#### Step 17. Run (⌘+R)

開啓後,若不在兩個TextField任一個輸入字串的話, 按兩個Button其中一個都可以印出整個Array裡面的資料



如果Key裡面輸入price就會在下面的TextView去找每個item的裡的price的資訊全部列印出來



如果Format裡面輸入year > 1950的話就會把year這個element大於1950的列印出來

