CS121 Tutorial 2

This is the second ios dev tutorial.

Purple bubbles give you information you'll need to know.

Yellow Bubbles tell you what to do.

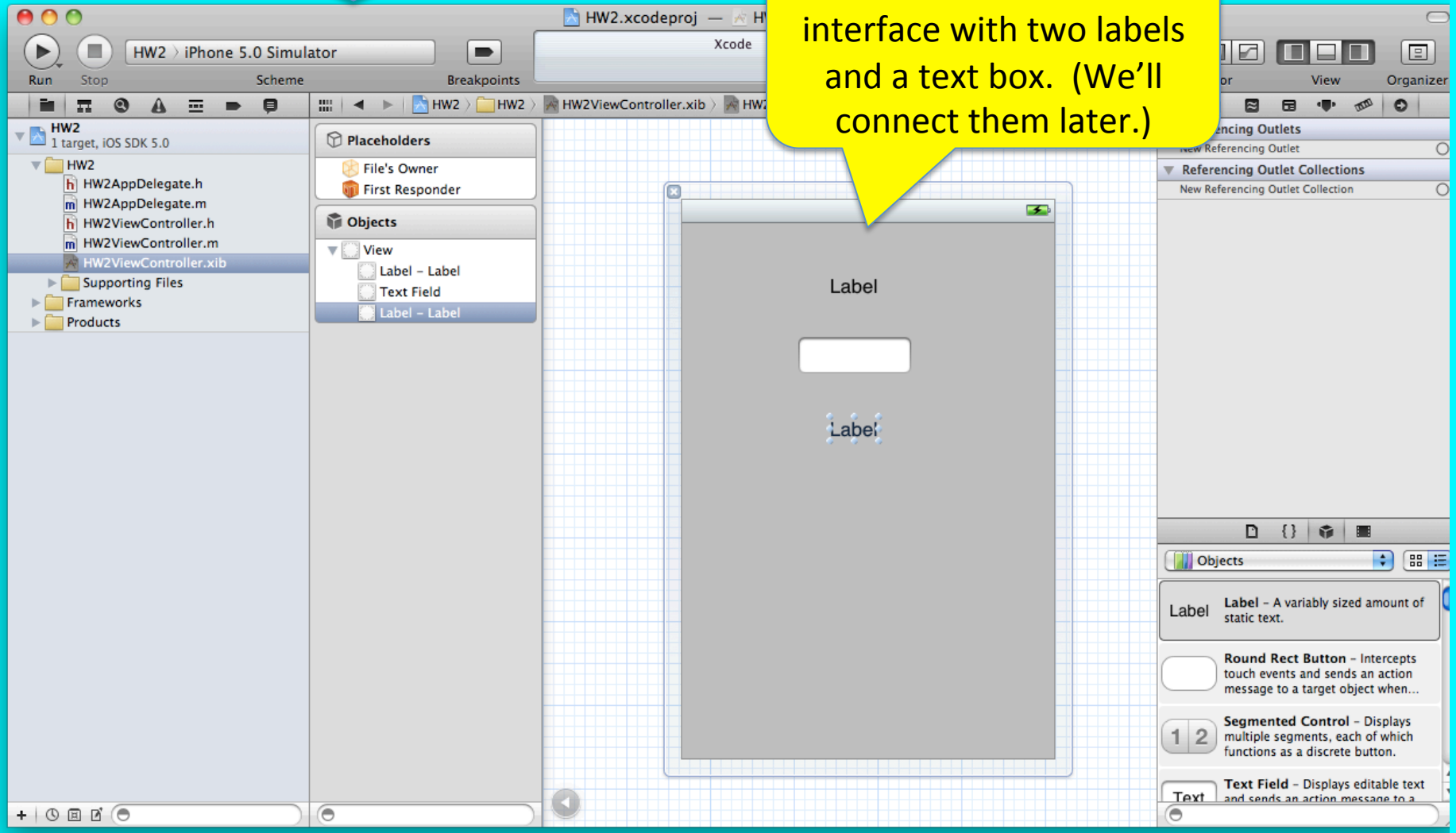
Orange bubbles tell you what you're not expected to understand yet. ☺

Goals for tutorial

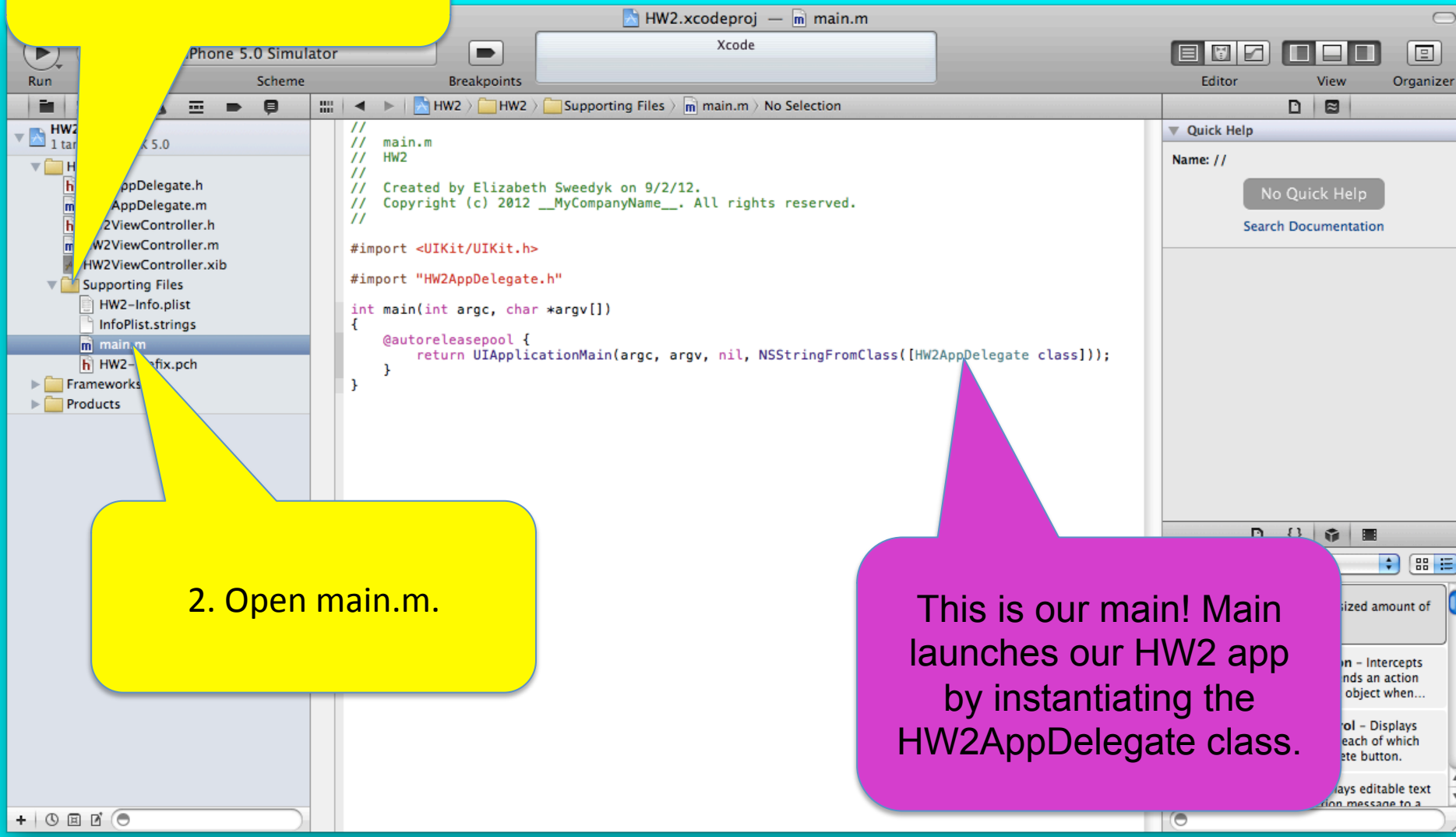
- Build the chopped fruit app
- Explore the architecture of an iOS app
- Explore some handy classes like NSArray and NSString

1. Follow the instructions from last time to create a new project named HW2.

2. Use the IB to create an interface with two labels and a text box. (We'll connect them later.)

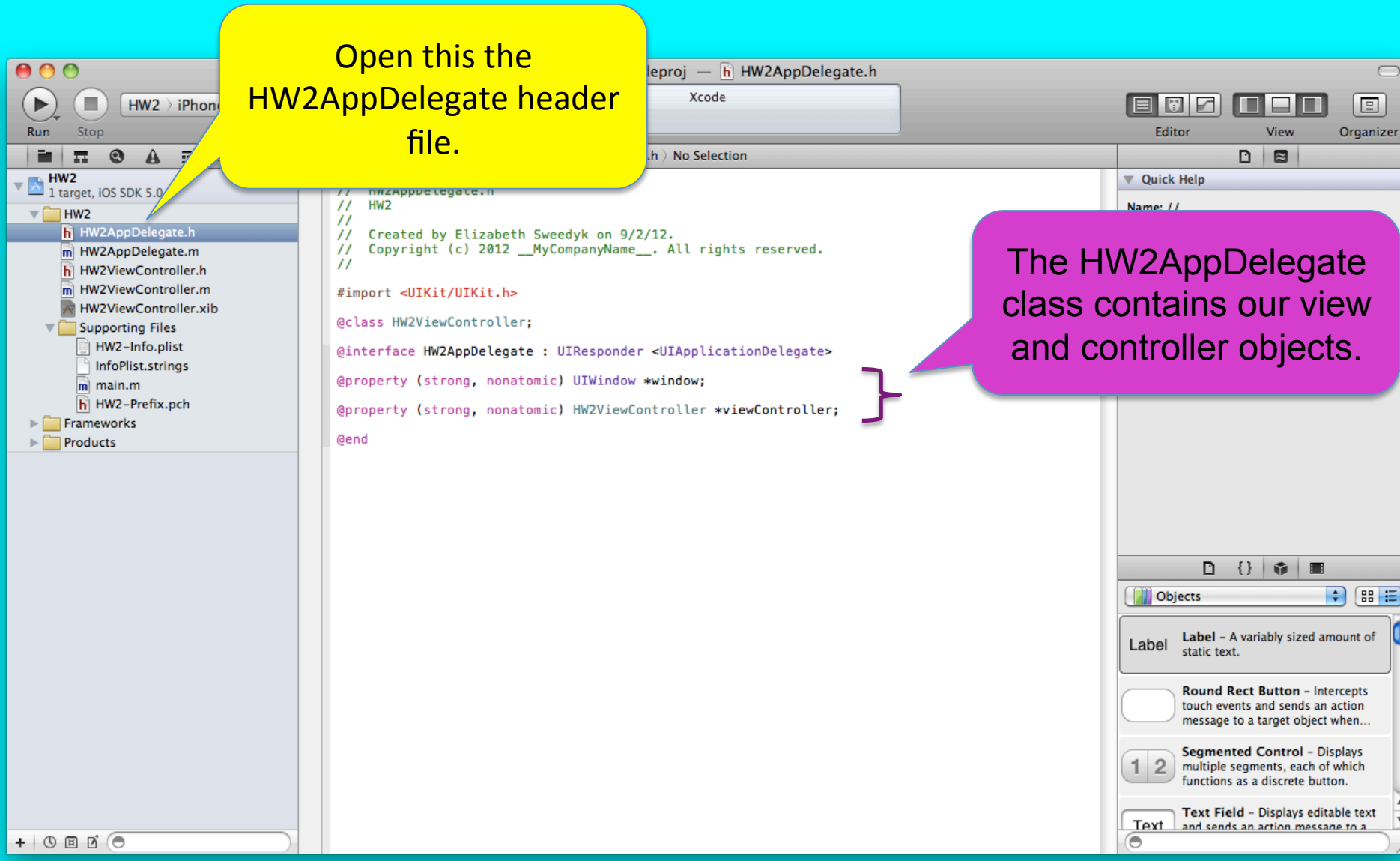


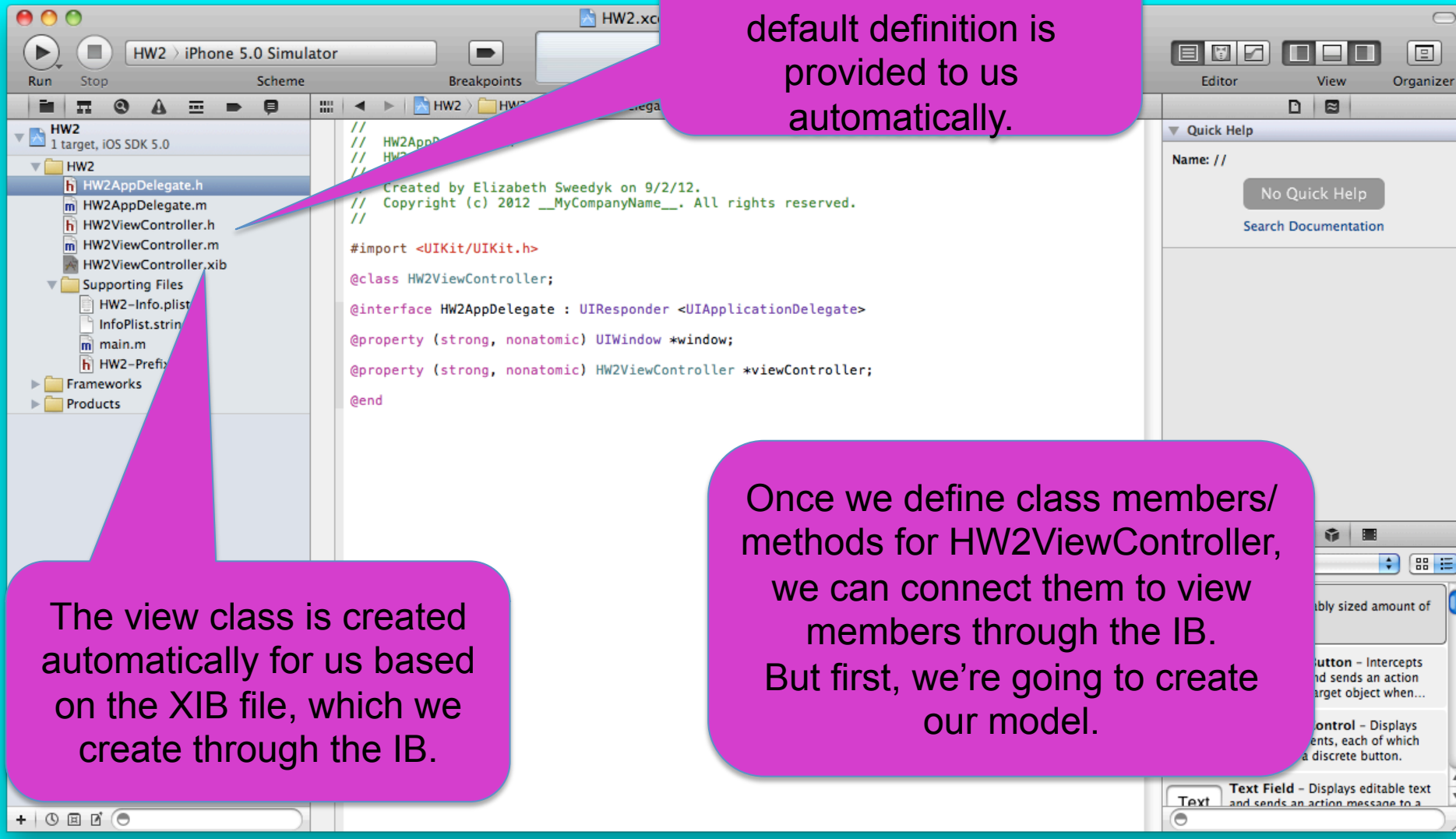
1. Open the Supporting Files folder.



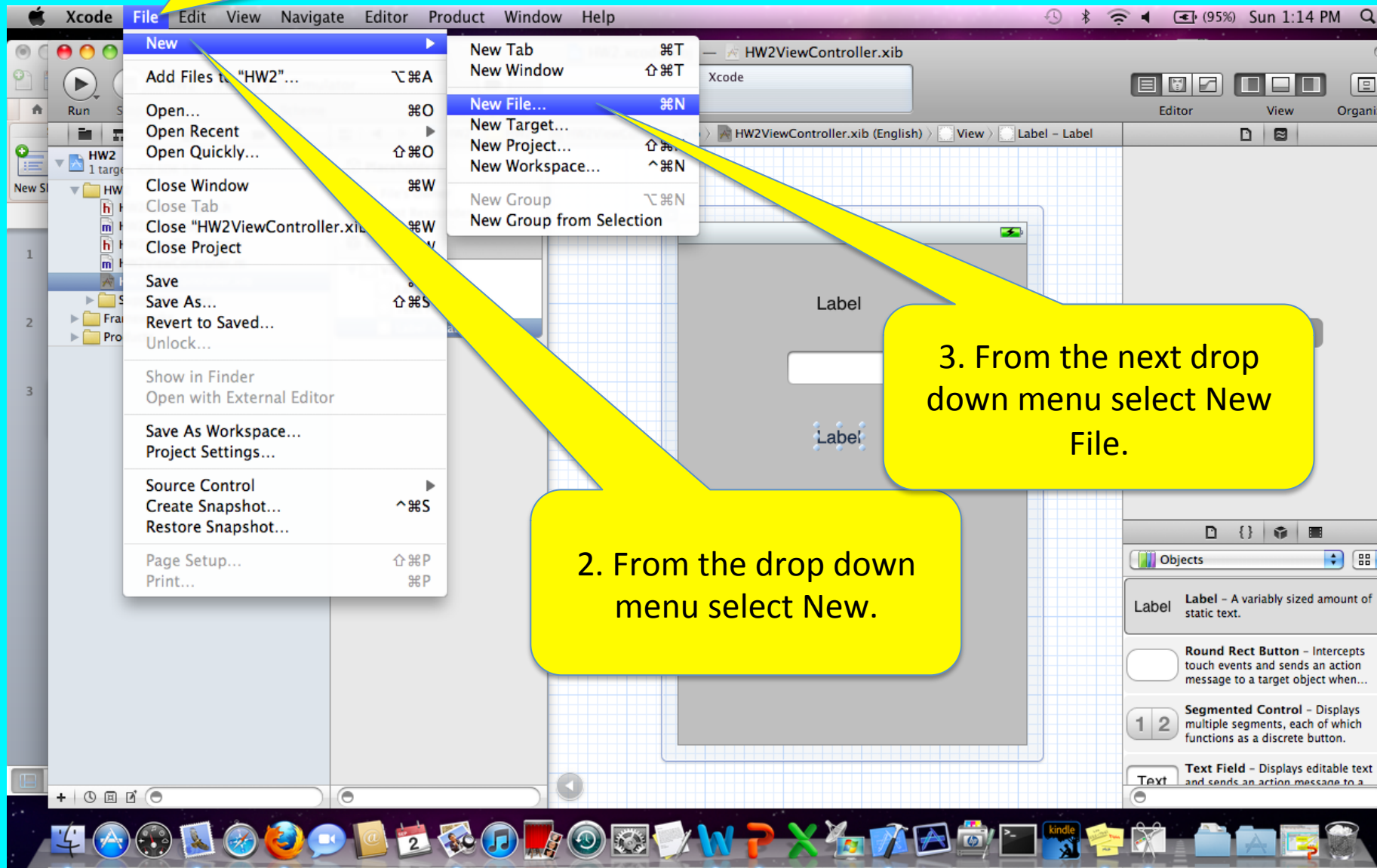
2. Open main.m.

This is our main! Main launches our HW2 app by instantiating the HW2AppDelegate class.


Open this the HW2AppDelegate header file.
HW2AppDelegate.h
HW2
1 target, iOS SDK 5.0
HW2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2ViewController.xib
Supporting Files
HW2-Info.plist
InfoPlist.strings
main.m
HW2-Prefix.pch
Frameworks
Products
No Selection
// HW2
// Created by Elizabeth Sweedyk on 9/2/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
#import <UIKit/UIKit.h>
@class HW2ViewController;
@interface HW2AppDelegate : UIResponder <UIApplicationDelegate>
@property (strong, nonatomic) UIWindow *window;
@property (strong, nonatomic) HW2ViewController *viewController;
@end
The HW2AppDelegate class contains our view and controller objects.
Quick Help
Objects
Label – A variably sized amount of static text.
Round Rect Button – Intercepts touch events and sends an action message to a target object when...
Segmented Control – Displays multiple segments, each of which functions as a discrete button.
Text Field – Displays editable text


The controller class is HW2ViewController. A default definition is provided to us automatically.
The view class is created automatically for us based on the XIB file, which we create through the IB.
Once we define class members/ methods for HW2ViewController, we can connect them to view members through the IB.
But first, we're going to create our model.
HW2
iPhone 5.0 Simulator
Run
Stop
Scheme
Breakpoints
Editor
View
Organizer
HW2
1 target, iOS SDK 5.0
HW2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2ViewController.xib
Supporting Files
HW2-Info.plist
InfoPlist.strin
main.m
HW2-Prefix
Frameworks
Products
Quick Help
Name: //
No Quick Help
Search Documentation
Created by Elizabeth Sweedyk on 9/2/12.
Copyright (c) 2012 __MyCompanyName__. All rights reserved.
#import <UIKit/UIKit.h>
@class HW2ViewController;
@interface HW2AppDelegate : UIResponder <UIApplicationDelegate>
@property (strong, nonatomic) UIWindow *window;
@property (strong, nonatomic) HW2ViewController *viewController;
@end
Text Field - Displays editable text

1. Select the File tab.

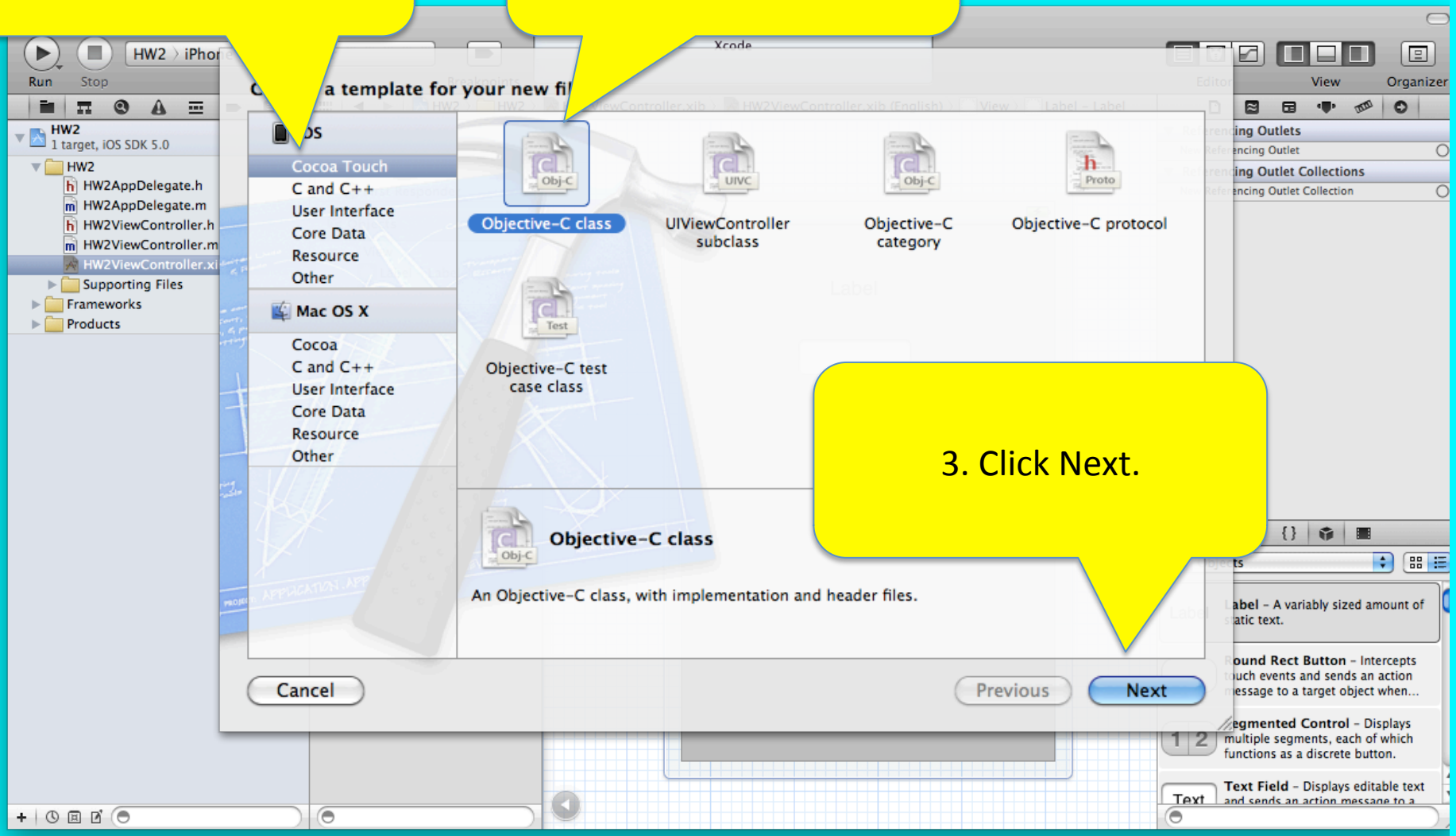


2. From the drop down menu select New.

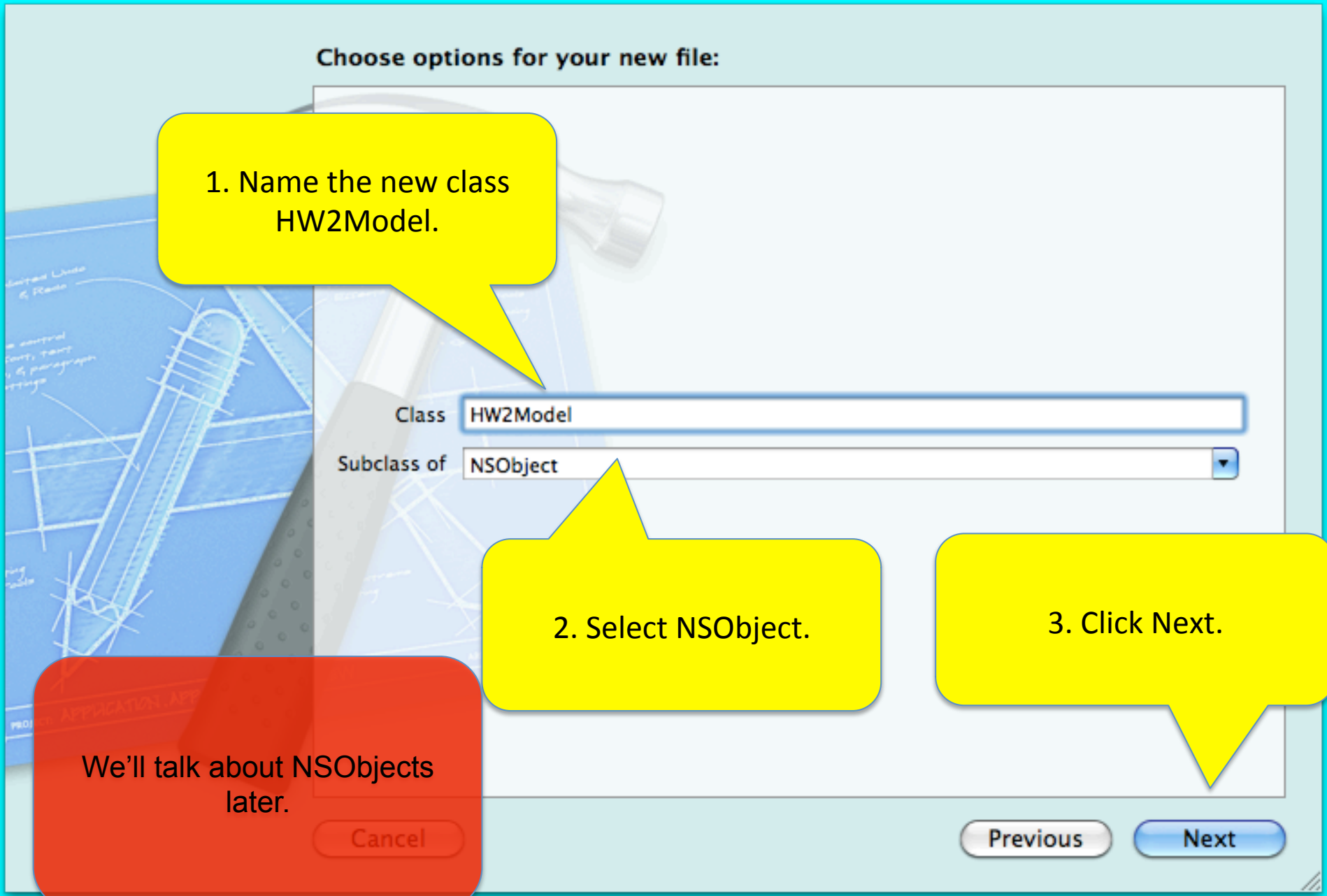
3. From the next drop down menu select New File.

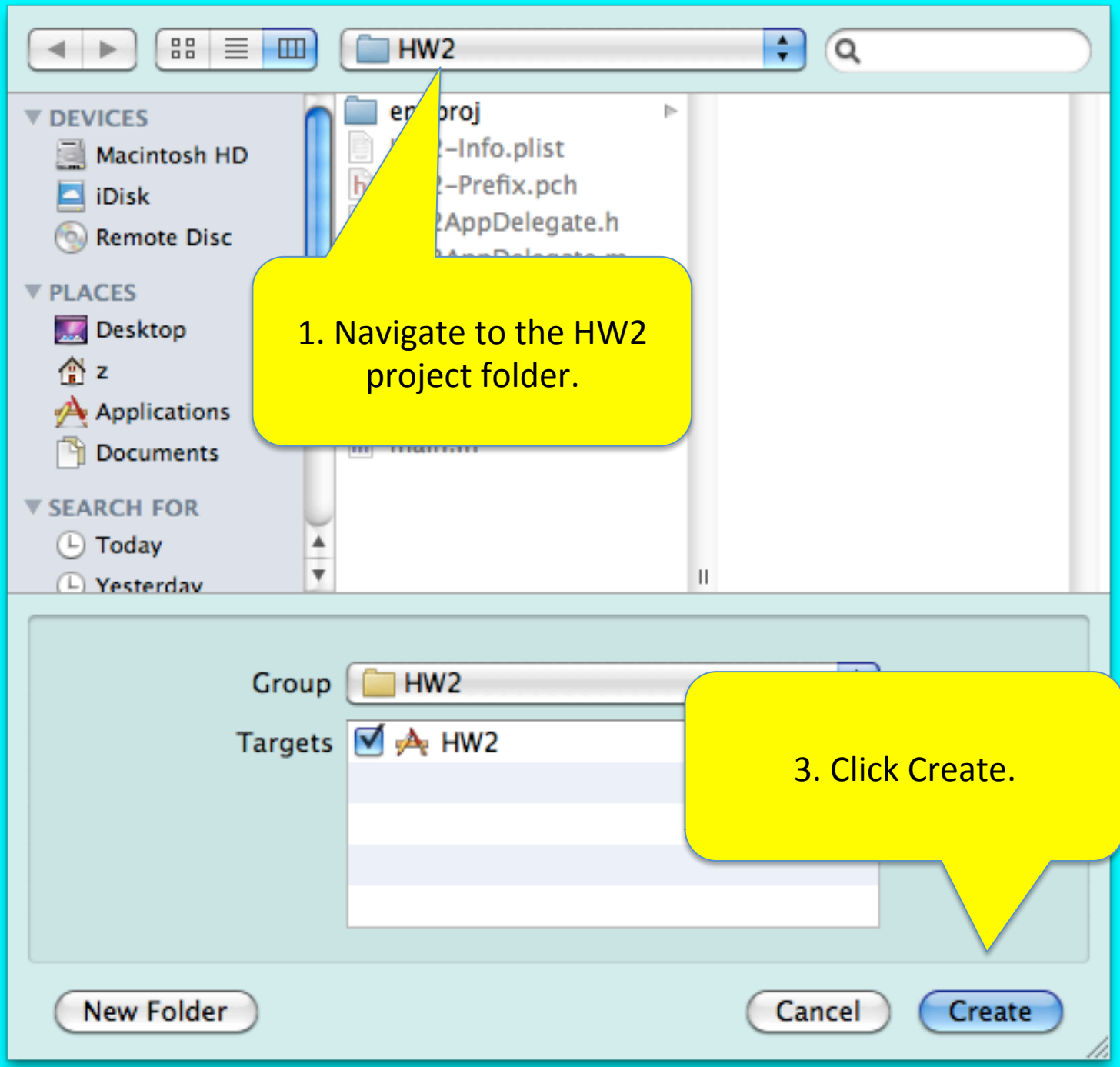
1. Select Cocoa Tools.

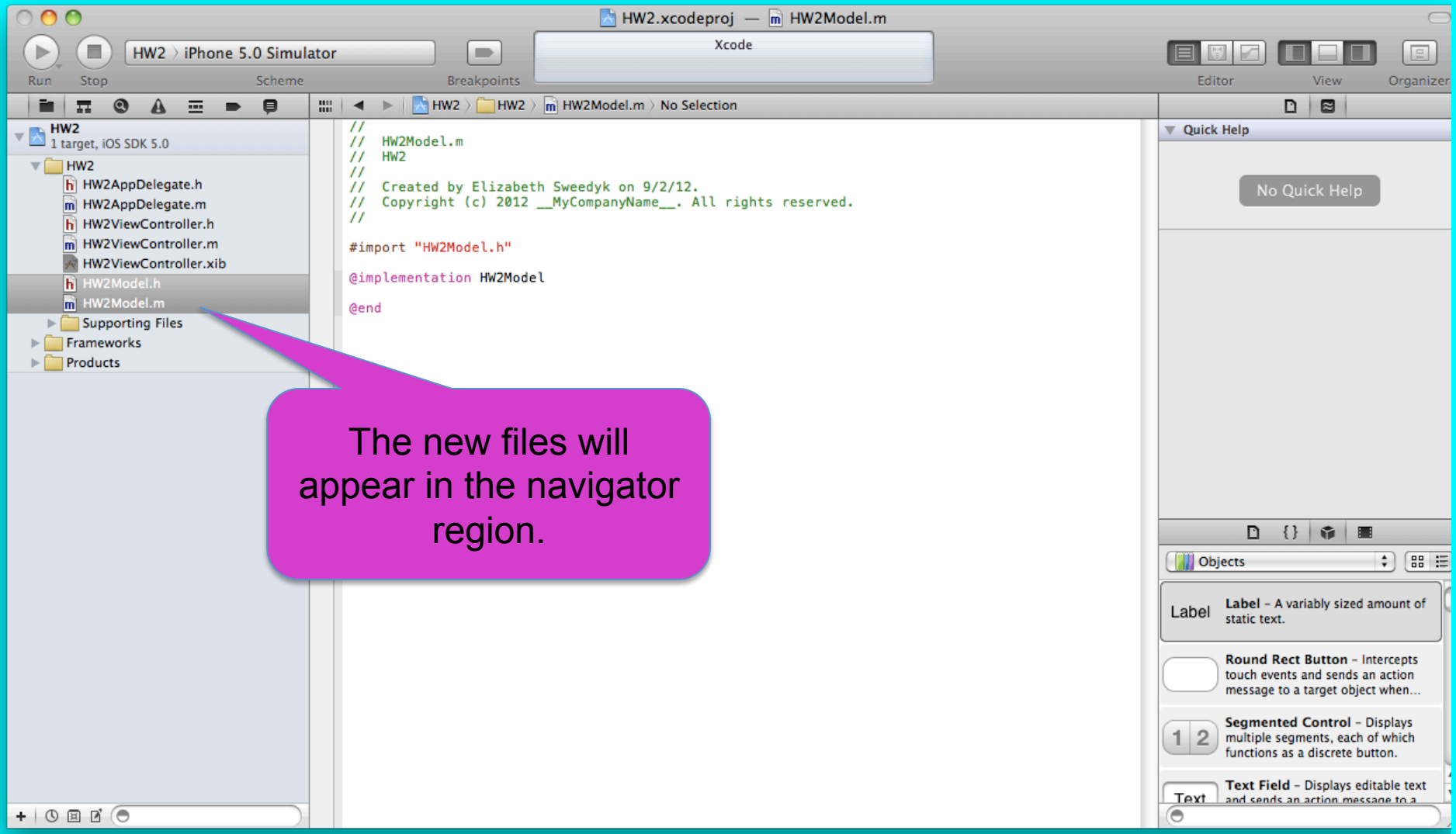
2. Select Objective-C class.

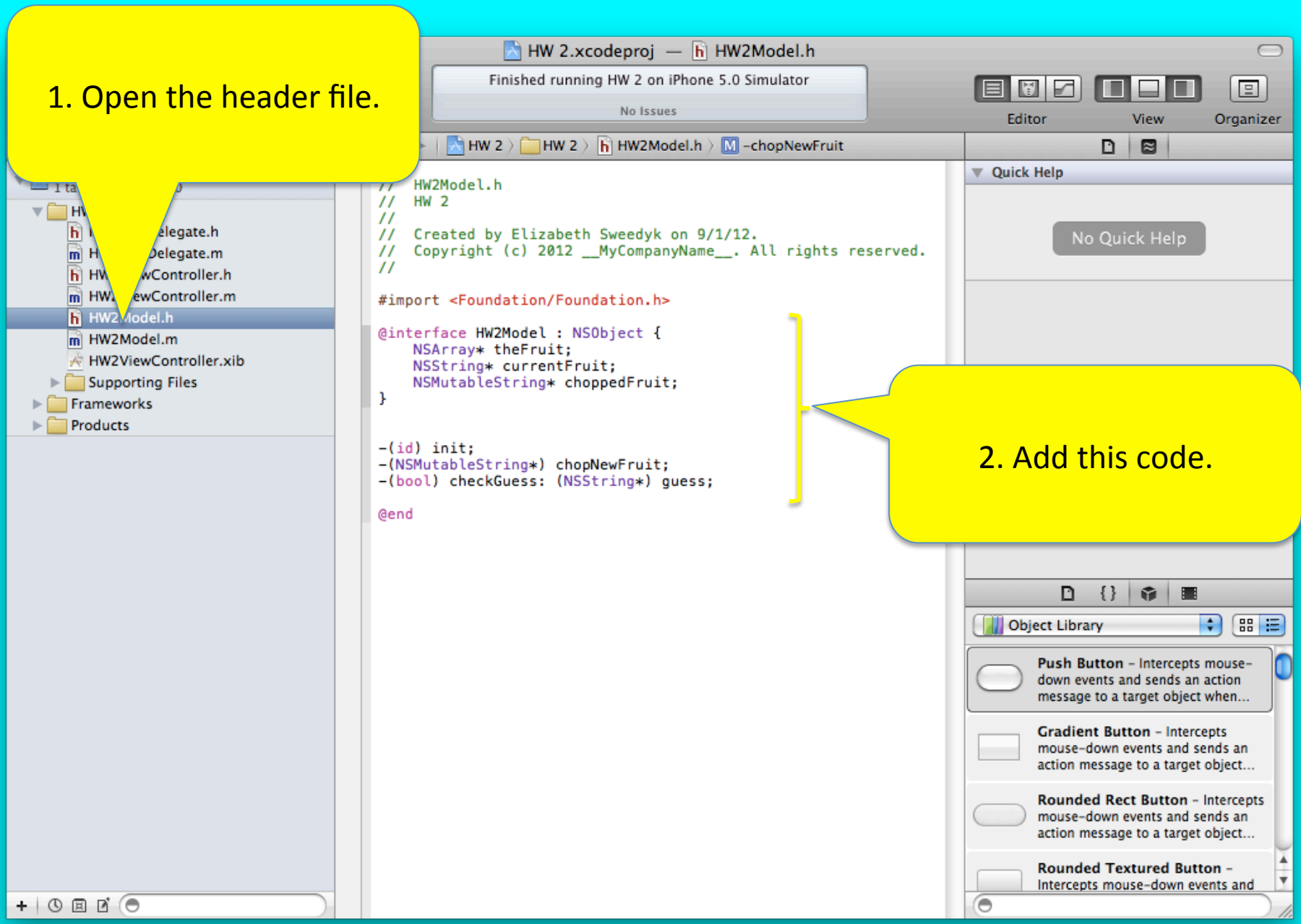


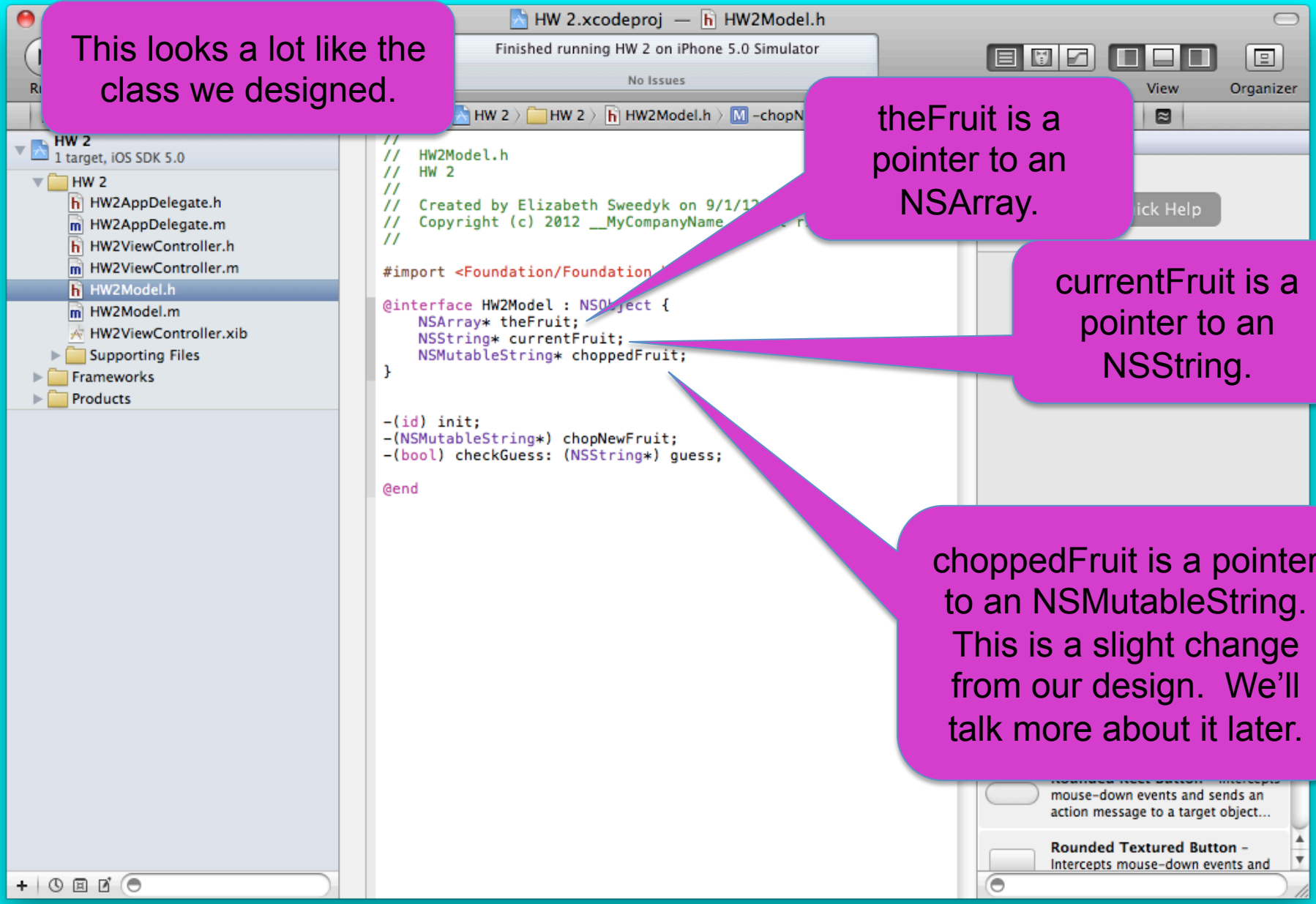
3. Click Next.

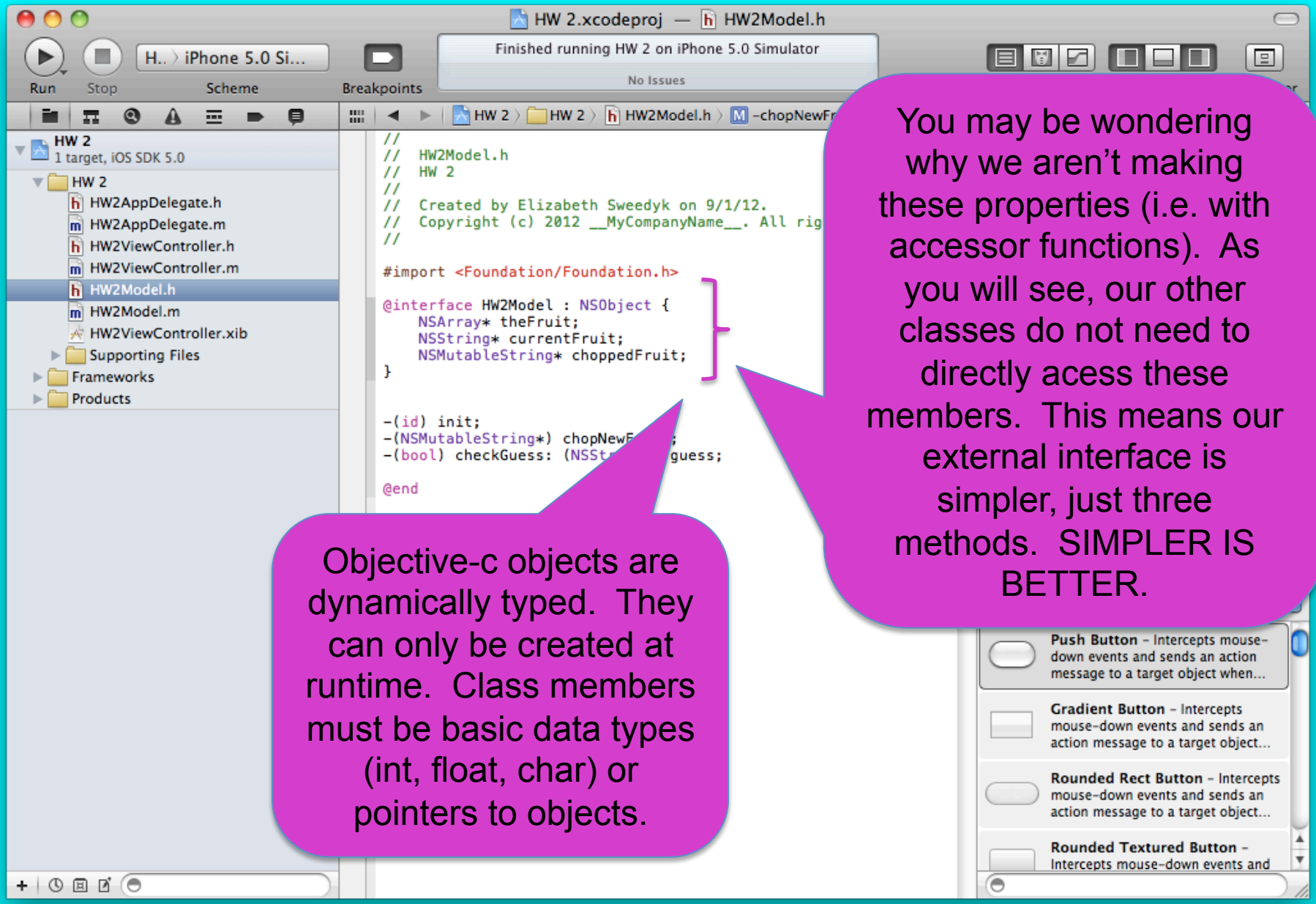

Choose options for your new file:
1. Name the new class HW2Model.
Class
HW2Model
Subclass of
NSObject
2. Select NSObject.
3. Click Next.
We'll talk about NSObjects later.
Cancel
Previous
Next


HW2
DEVICES
Macintosh HD
iDisk
Remote Disc
PLACES
Desktop
z
Applications
Documents
SEARCH FOR
Today
Yesterday
proj
-Info.plist
-Prefix.pch
AppDelegate.h
1. Navigate to the HW2 project folder.
Group
HW2
Targets
HW2
3. Click Create.
New Folder
Cancel
Create


HW2.xcodeproj — HW2Model.m
Run
Stop
HW2 › iPhone 5.0 Simulator
Scheme
Breakpoints
Xcode
Editor
View
Organizer
HW2
1 target, iOS SDK 5.0
HW2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2ViewController.xib
HW2Model.h
HW2Model.m
Supporting Files
Frameworks
Products
HW2 › HW2 › HW2Model.m › No Selection
//
// HW2Model.m
// HW2
//
// Created by Elizabeth Sweedyk on 9/2/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
//
#import "HW2Model.h"
@implementation HW2Model
@end
Quick Help
No Quick Help
Objects
Label – A variably sized amount of static text.
Round Rect Button – Intercepts touch events and sends an action message to a target object when...
Segmented Control – Displays multiple segments, each of which functions as a discrete button.
Text Field – Displays editable text
The new files will appear in the navigator region.

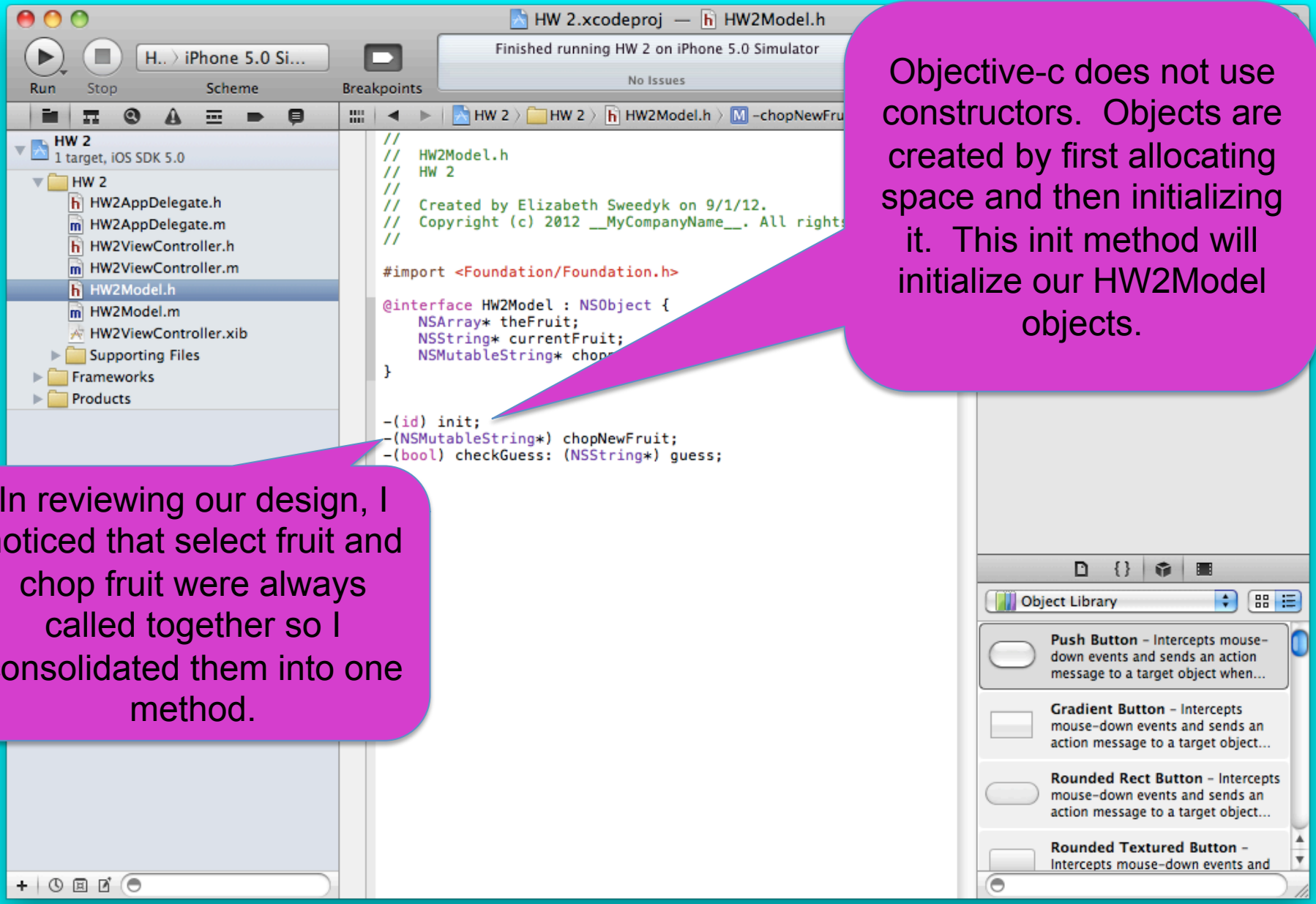

1. Open the header file.
HW 2.xcodeproj — HW2Model.h
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2 › HW 2 › HW2Model.h › -chopNewFruit
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
// HW2Model.h
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
//
#import <Foundation/Foundation.h>
@interface HW2Model : NSObject {
NSArray* theFruit;
NSString* currentFruit;
NSMutableString* choppedFruit;
}
-(id) init;
-(NSMutableString*) chopNewFruit;
-(bool) checkGuess: (NSString*) guess;
@end
2. Add this code.
Quick Help
No Quick Help
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

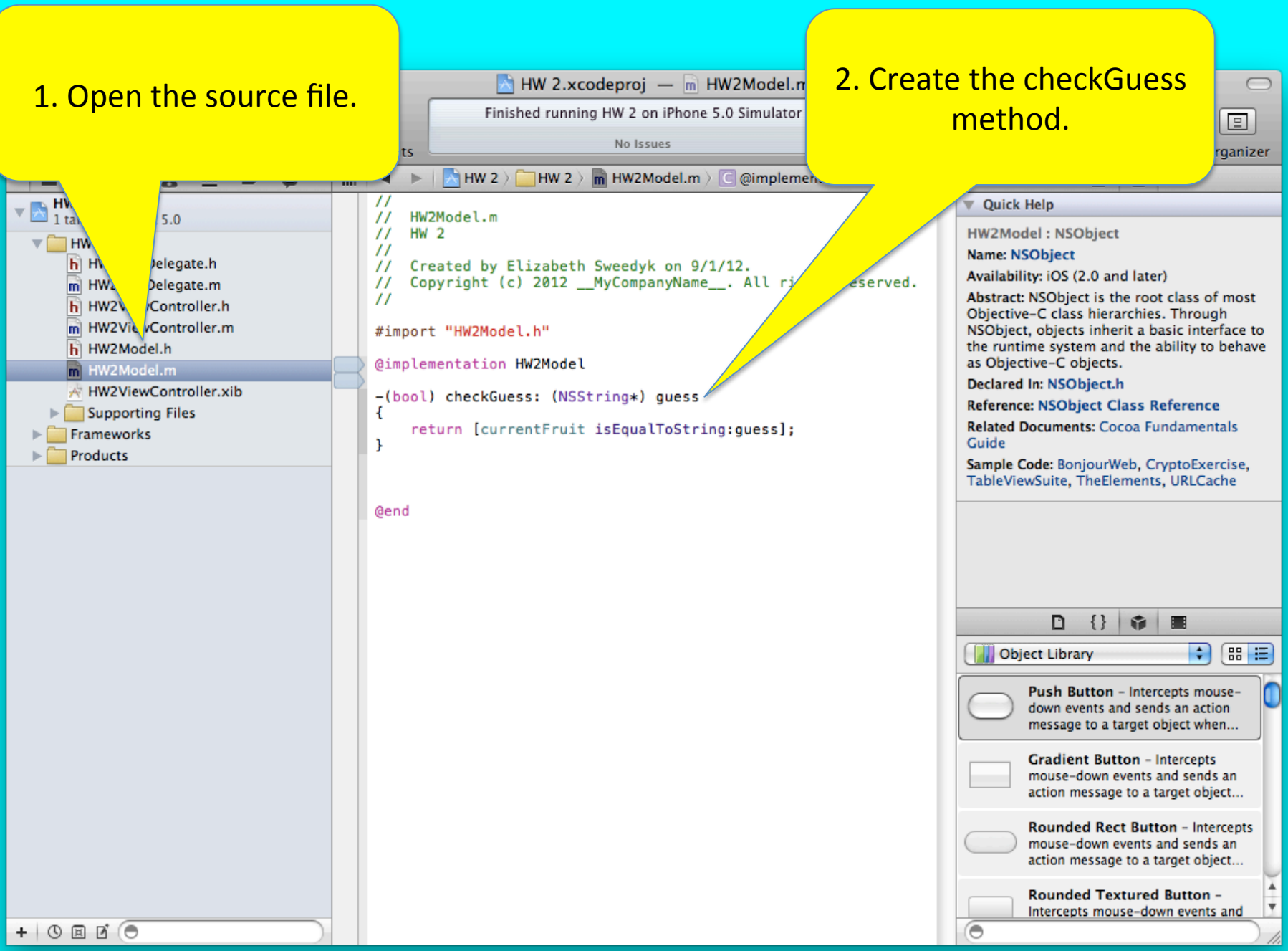

This looks a lot like the class we designed.
HW 2.xcodeproj — HW2Model.h
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 > HW 2 > HW2Model.h
//
// HW2Model.h
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12
// Copyright (c) 2012 __MyCompanyName
//
#import <Foundation/Foundation
@interface HW2Model : NSObject {
NSArray* theFruit;
NSString* currentFruit;
NSMutableString* choppedFruit;
}
-(id) init;
-(NSMutableString*) chopNewFruit;
-(bool) checkGuess: (NSString*) guess;
@end
theFruit is a pointer to an NSArray.
currentFruit is a pointer to an NSString.
choppedFruit is a pointer to an NSMutableString. This is a slight change from our design. We'll talk more about it later.
mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

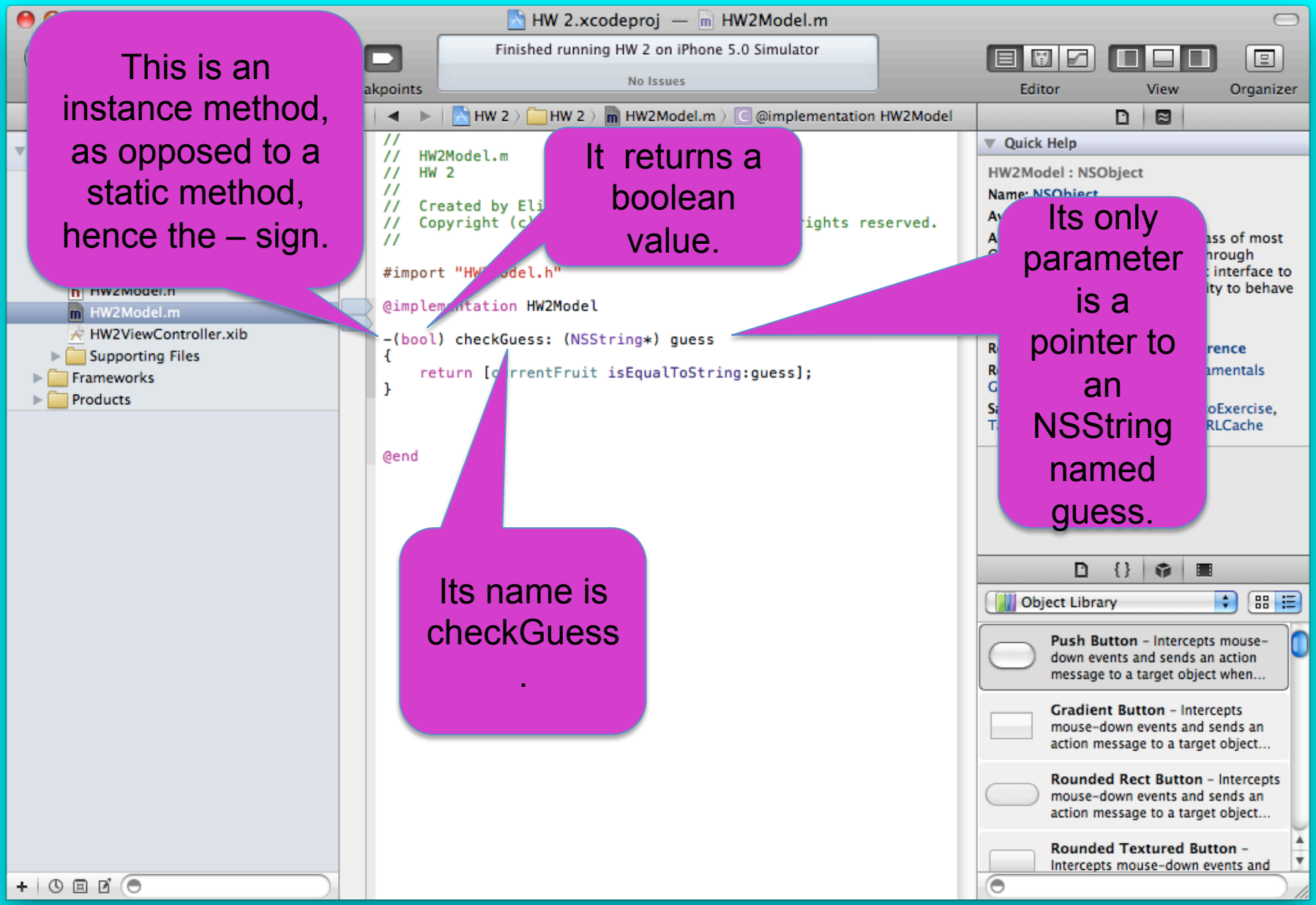


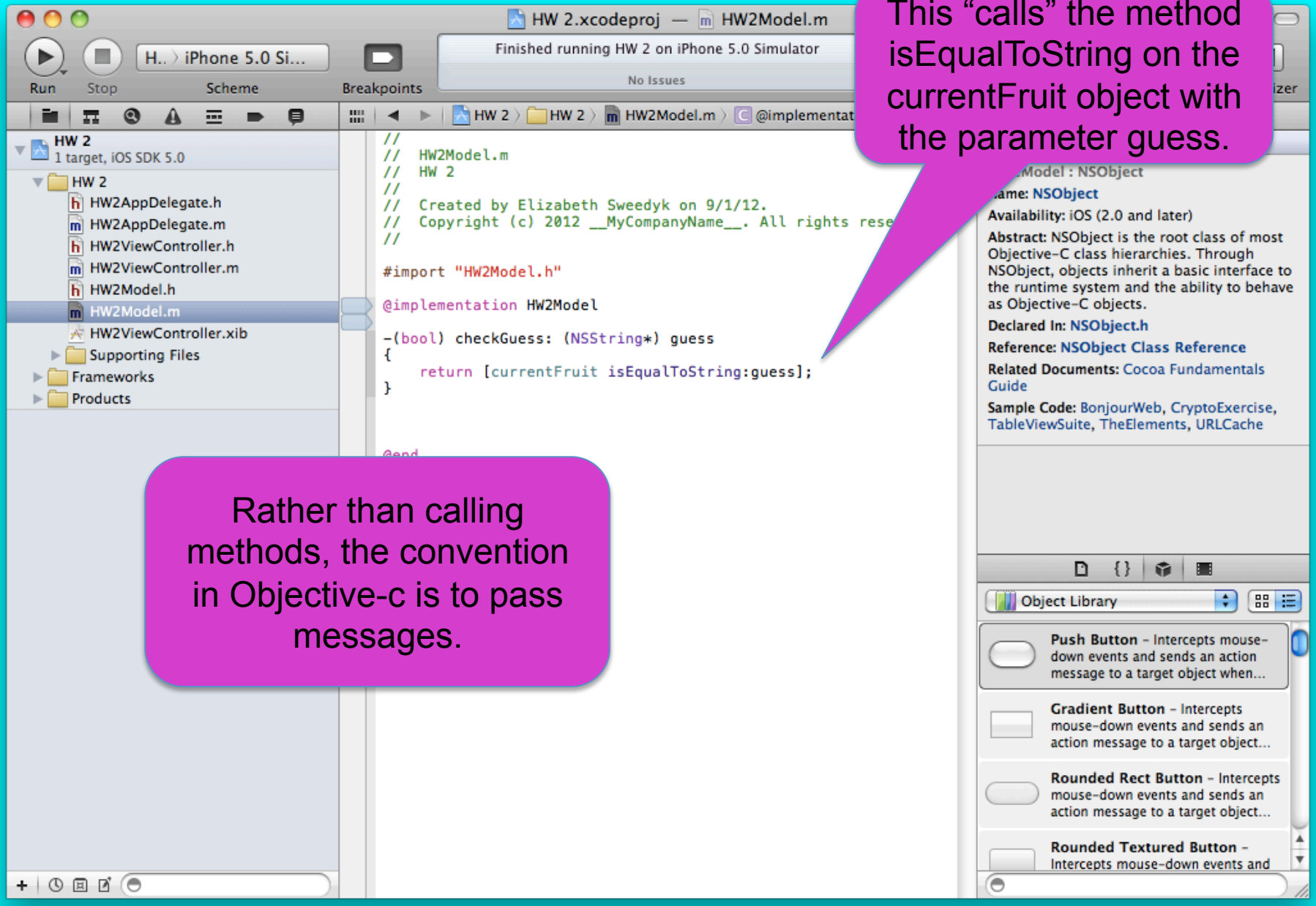
You may be wondering why we aren't making these properties (i.e. with accessor functions). As you will see, our other classes do not need to directly acess these members. This means our external interface is simpler, just three methods. SIMPLER IS BETTER.

Objective-c objects are dynamically typed. They can only be created at runtime. Class members must be basic data types (int, float, char) or pointers to objects.


HW 2.xcodeproj — HW2Model.h
Run
Stop
Scheme
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
//
// HW2Model.h
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights
//
#import <Foundation/Foundation.h>
@interface HW2Model : NSObject {
NSArray* theFruit;
NSString* currentFruit;
NSMutableString* chop
}
-(id) init;
-(NSMutableString*) chopNewFruit;
-(bool) checkGuess: (NSString*) guess;
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and
Objective-c does not use constructors. Objects are created by first allocating space and then initializing it. This init method will initialize our HW2Model objects.
In reviewing our design, I noticed that select fruit and chop fruit were always called together so I consolidated them into one method.


1. Open the source file.
2. Create the checkGuess method.
HW 2.xcodeproj — HW2Model.m
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
HW 2 › HW 2 › HW2Model.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All ri eserved.
//
#import "HW2Model.h"
@implementation HW2Model
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
Quick Help
HW2Model : NSObject
Name: NSObject
Availability: iOS (2.0 and later)
Abstract: NSObject is the root class of most Objective-C class hierarchies. Through NSObject, objects inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.
Declared In: NSObject.h
Reference: NSObject Class Reference
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and


This is an instance method, as opposed to a static method, hence the – sign.
It returns a boolean value.
Its only parameter is a pointer to an NSString named guess.
Its name is checkGuess .
HW 2.xcodeproj — HW2Model.m
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2 > HW 2 > HW2Model.m > @implementation HW2Model
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
Quick Help
HW2Model : NSObject
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and


This “calls” the method isEqualToString on the currentFruit object with the parameter guess.
Rather than calling methods, the convention in Objective-c is to pass messages.
HW 2.xcodeproj — HW2Model.m
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Run
Stop
Scheme
Breakpoints
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights rese
//
#import "HW2Model.h"
@implementation HW2Model
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
Model : NSObject
ame: NSObject
Availability: iOS (2.0 and later)
Abstract: NSObject is the root class of most Objective-C class hierarchies. Through NSObject, objects inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.
Declared In: NSObject.h
Reference: NSObject Class Reference
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

C++ method call:
Obj->method(arg);

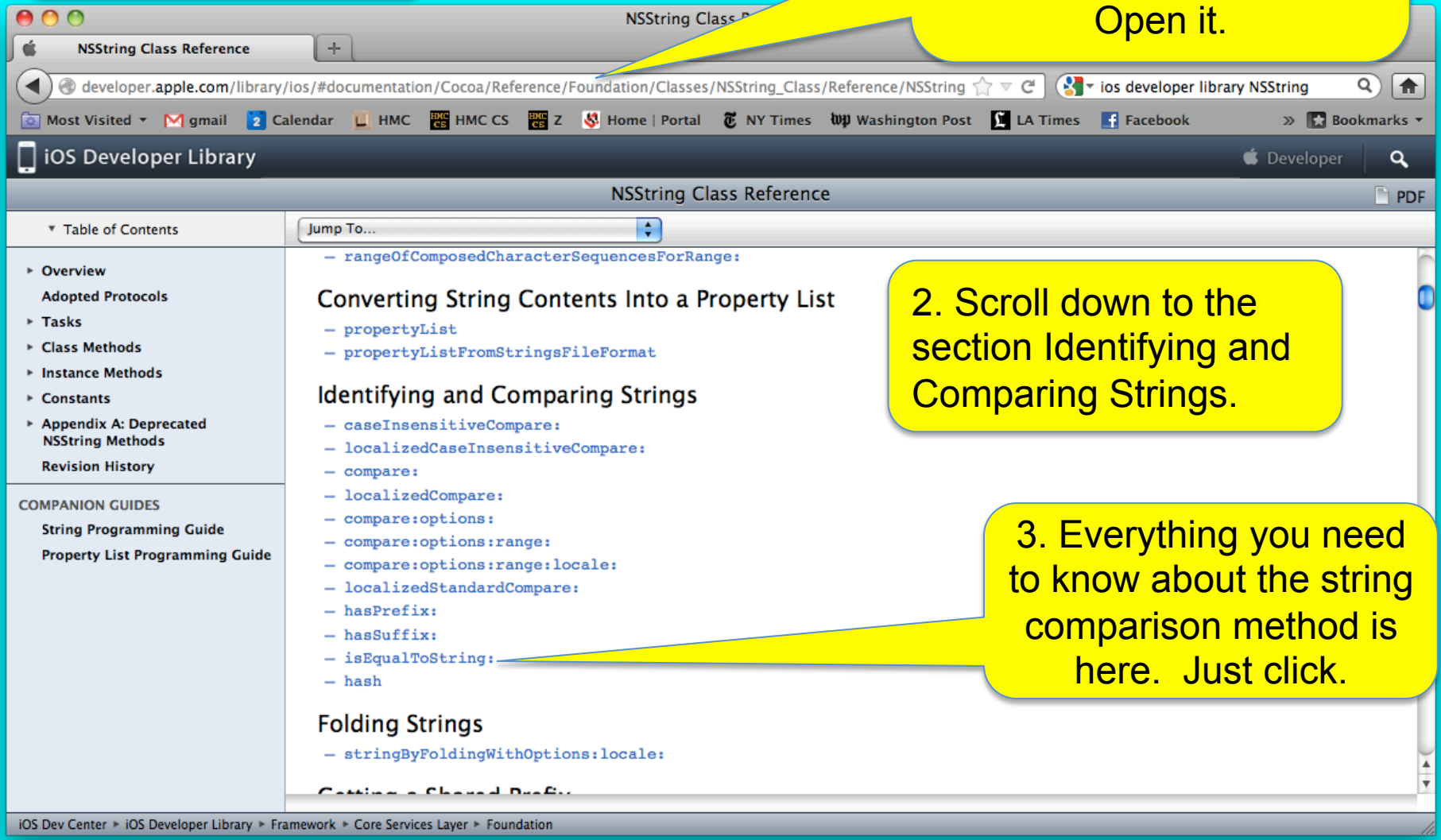
is equivalent to

Objective-c message passing:
[Obj method:arg];

What about methods with multiple arguments? Hold that thought.

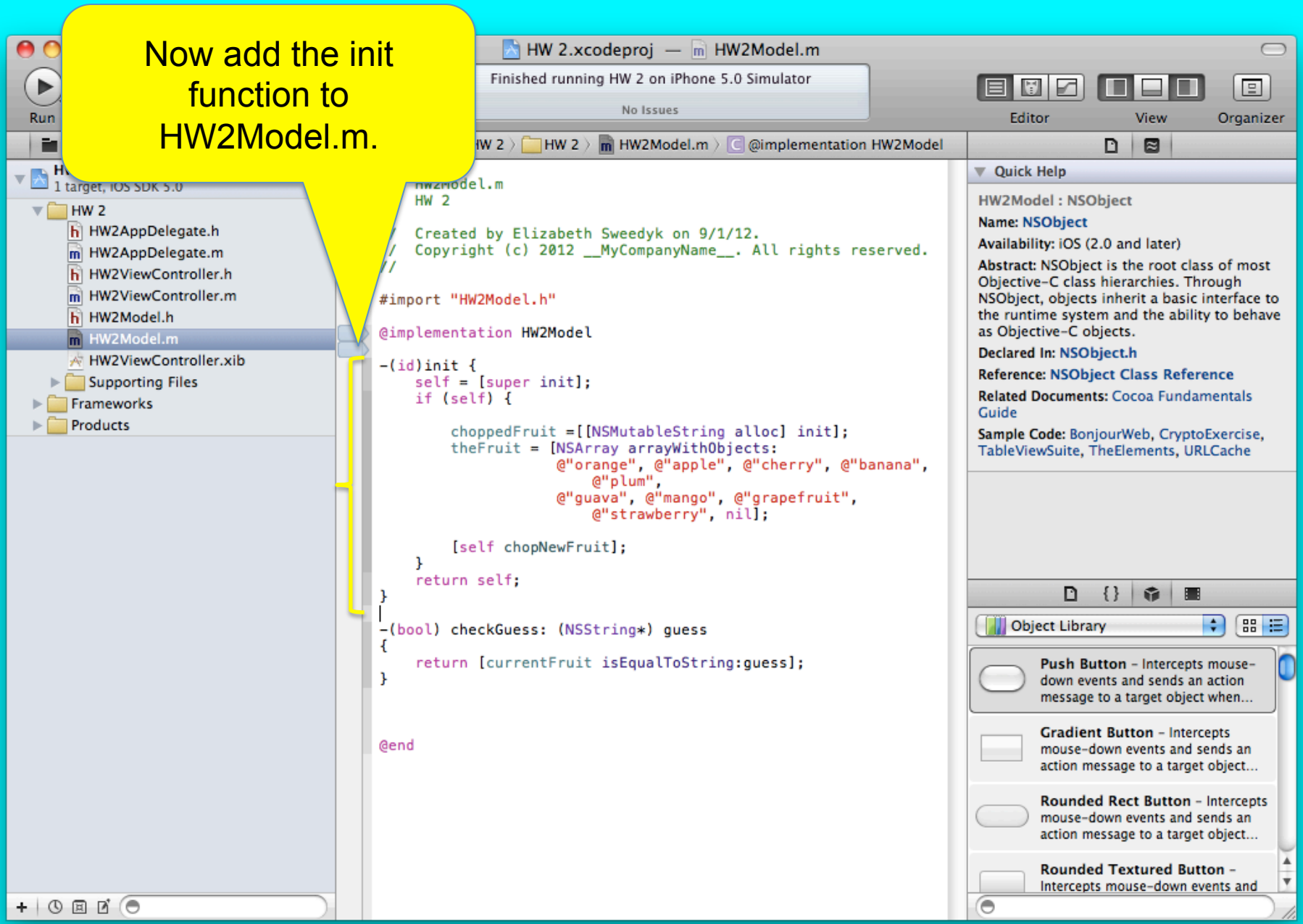
Now is a good time to talk about online documentation.

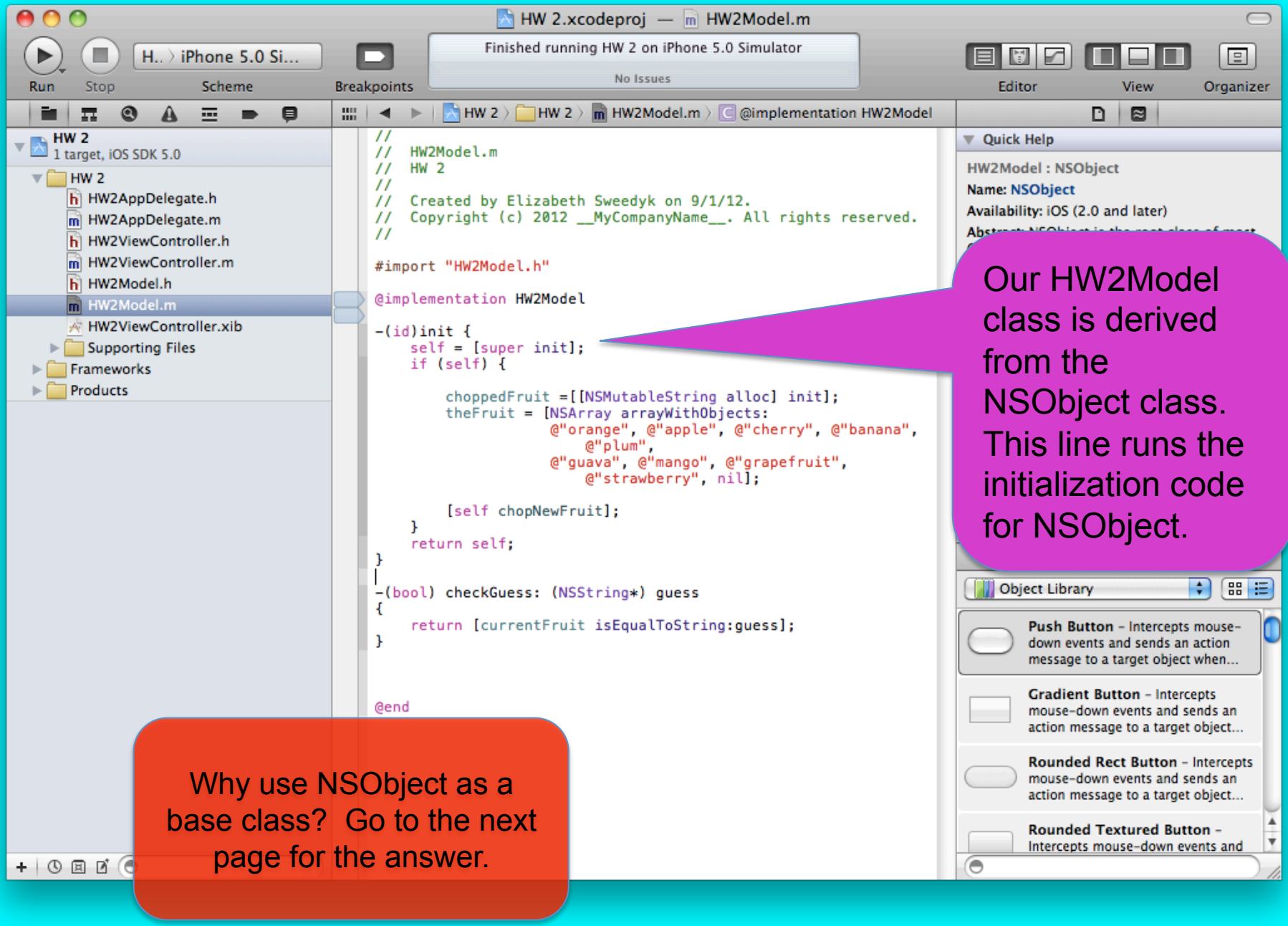
1. Open a web browser and google ios developer NSString. One of the first links will be this one. Open it.



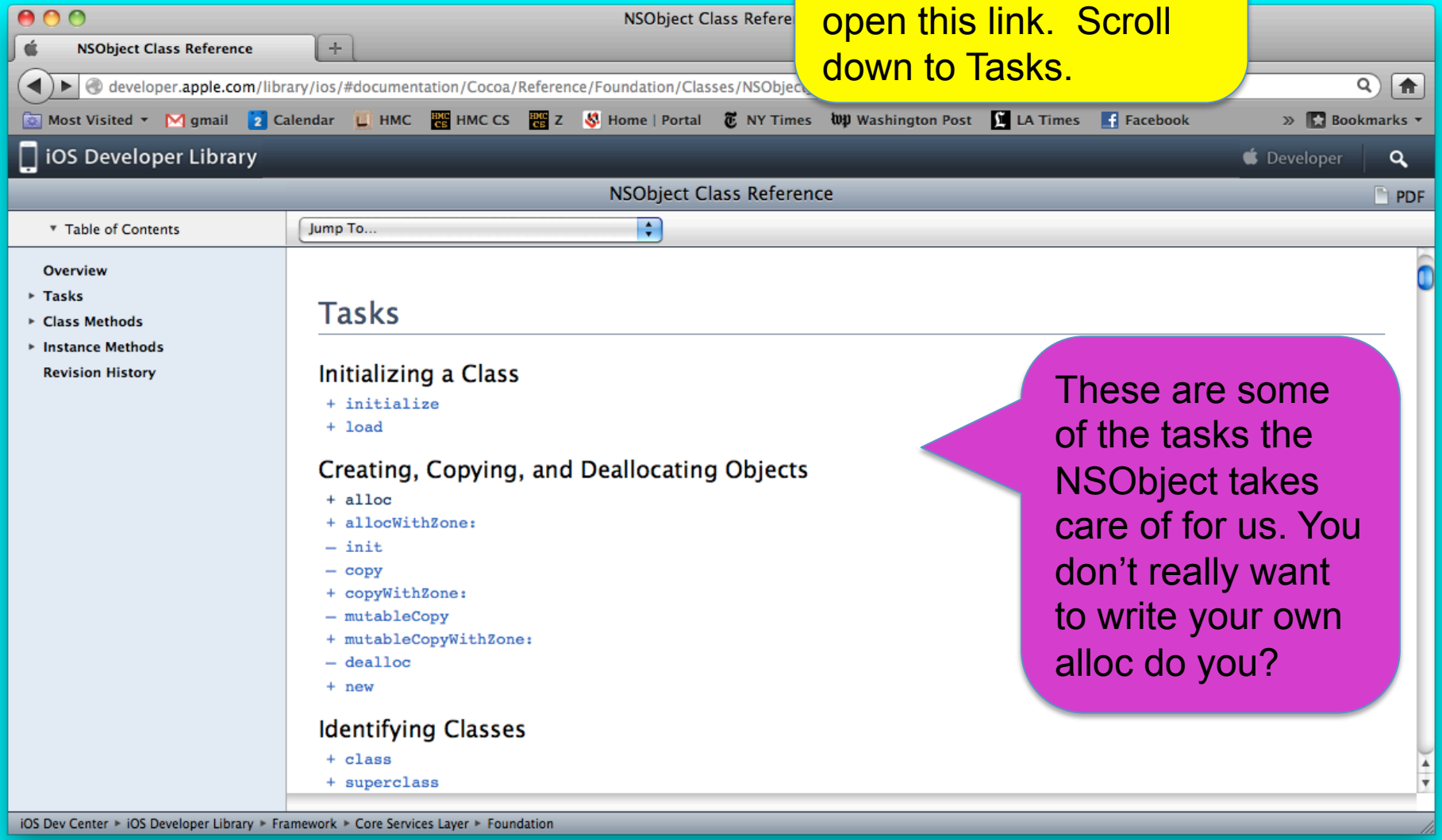
2. Scroll down to the section Identifying and Comparing Strings.

3. Everything you need to know about the string comparison method is here. Just click.


Now add the init function to HW2Model.m.
HW 2.xcodeproj — HW2Model.m
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Run
Editor
View
Organizer
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW2Model.m
HW 2
Created by Elizabeth Sweedyk on 9/1/12.
Copyright (c) 2012 __MyCompanyName__. All rights reserved.
#import "HW2Model.h"
@implementation HW2Model
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableString alloc] init];
theFruit = [NSArray arrayWithObjects:
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
[self chopNewFruit];
}
return self;
}
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
Quick Help
HW2Model : NSObject
Name: NSObject
Availability: iOS (2.0 and later)
Abstract: NSObject is the root class of most Objective-C class hierarchies. Through NSObject, objects inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.
Declared In: NSObject.h
Reference: NSObject Class Reference
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

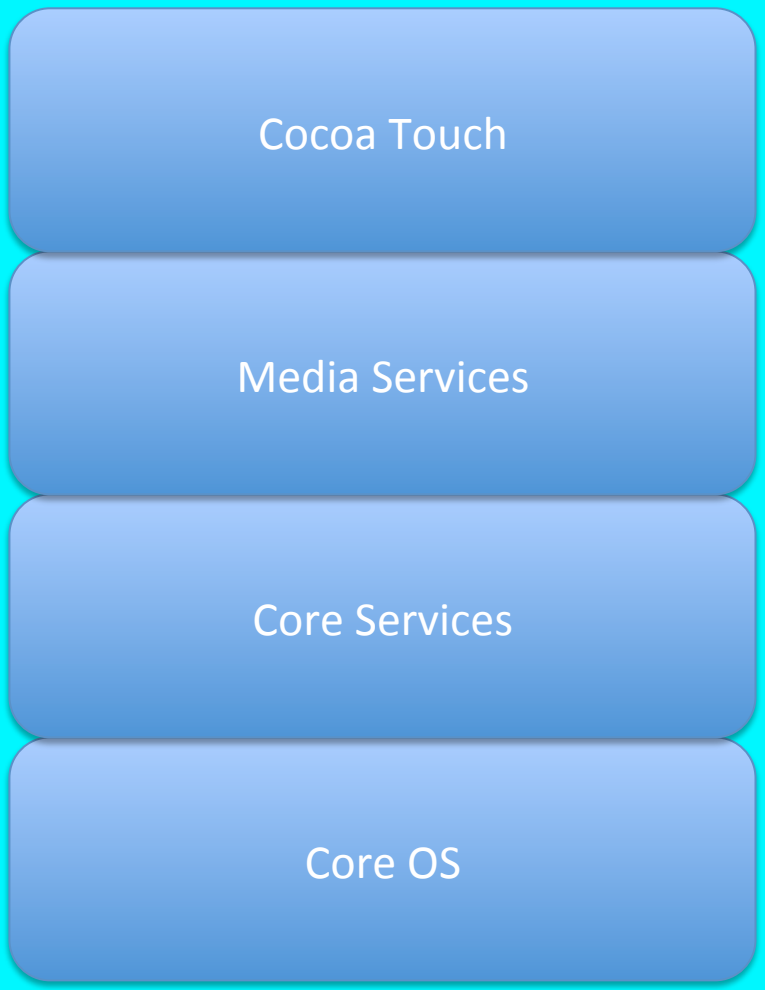

HW 2.xcodeproj — HW2Model.m
Run
Stop
Scheme
iPhone 5.0 Si...
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 > HW 2 > HW2Model.m > @implementation HW2Model
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
//
#import "HW2Model.h"
@implementation HW2Model
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableString alloc] init];
theFruit = [NSArray arrayWithObjects:
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
[self chopNewFruit];
}
return self;
}
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
Quick Help
HW2Model : NSObject
Name: NSObject
Availability: iOS (2.0 and later)
Our HW2Model class is derived from the NSObject class. This line runs the initialization code for NSObject.
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and
Why use NSObject as a base class? Go to the next page for the answer.

Open a browser, google “ios developer library NSObject” and open this link. Scroll down to Tasks.



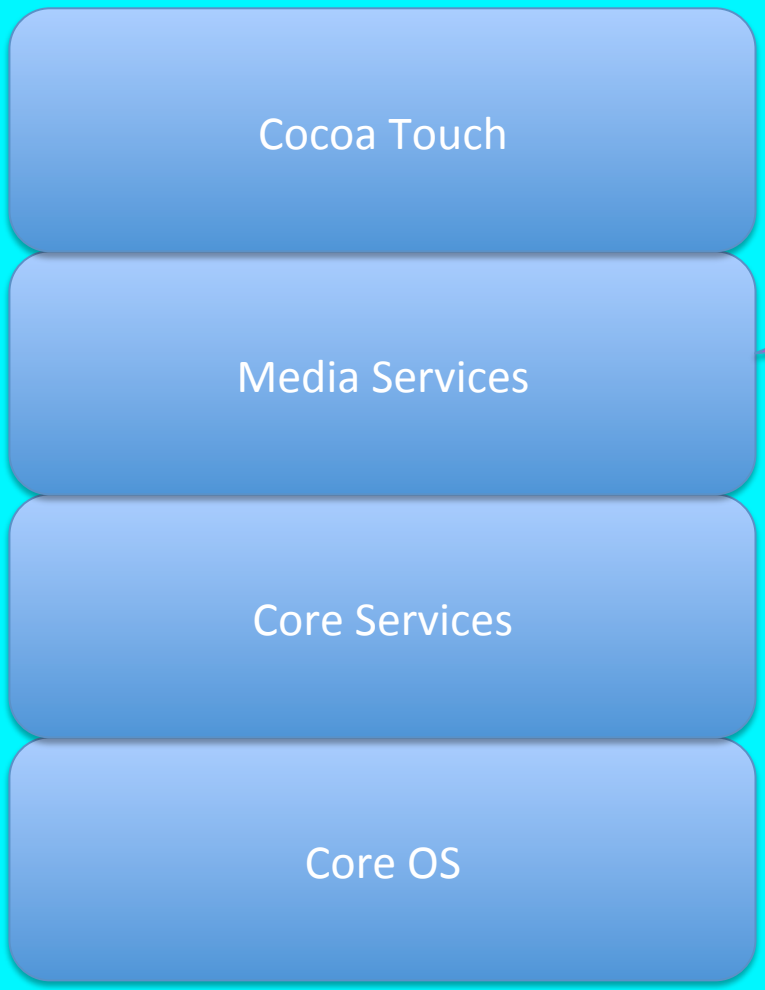
These are some of the tasks the NSObject takes care of for us. You don’t really want to write your own alloc do you?

The IOS Architecture has four layers that provide amazing services.



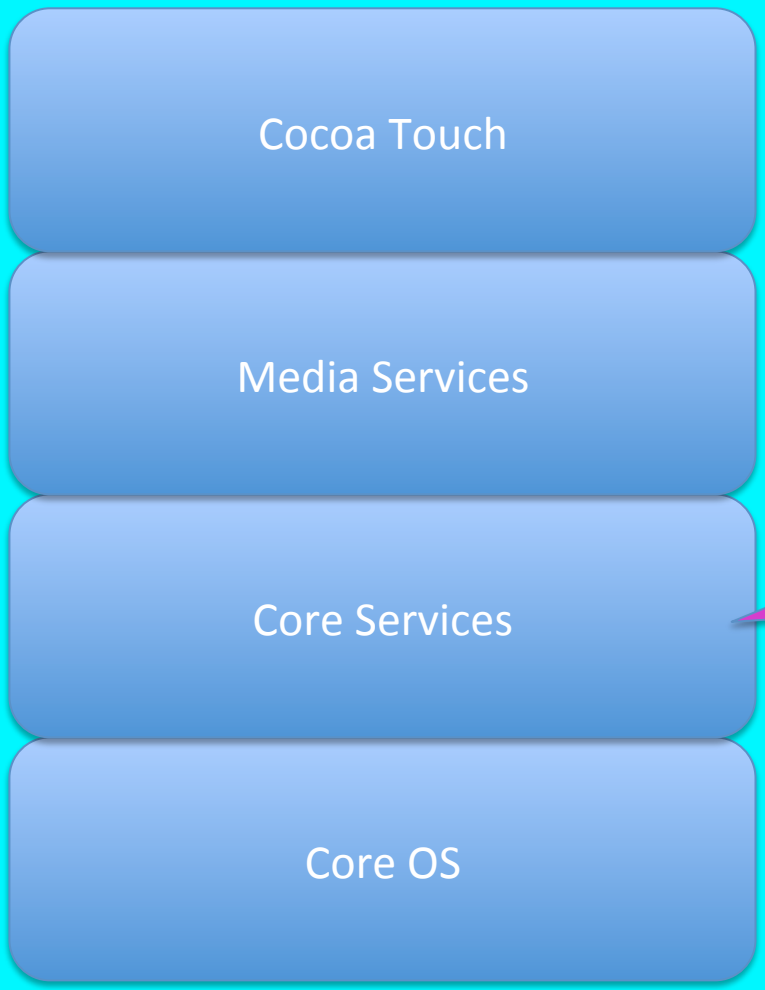
Cocoa Touch includes important frameworks for building iOS application. One is UIKit, which provides the UI elements we've been using.

The IOS Architecture has four layers that provide amazing services.



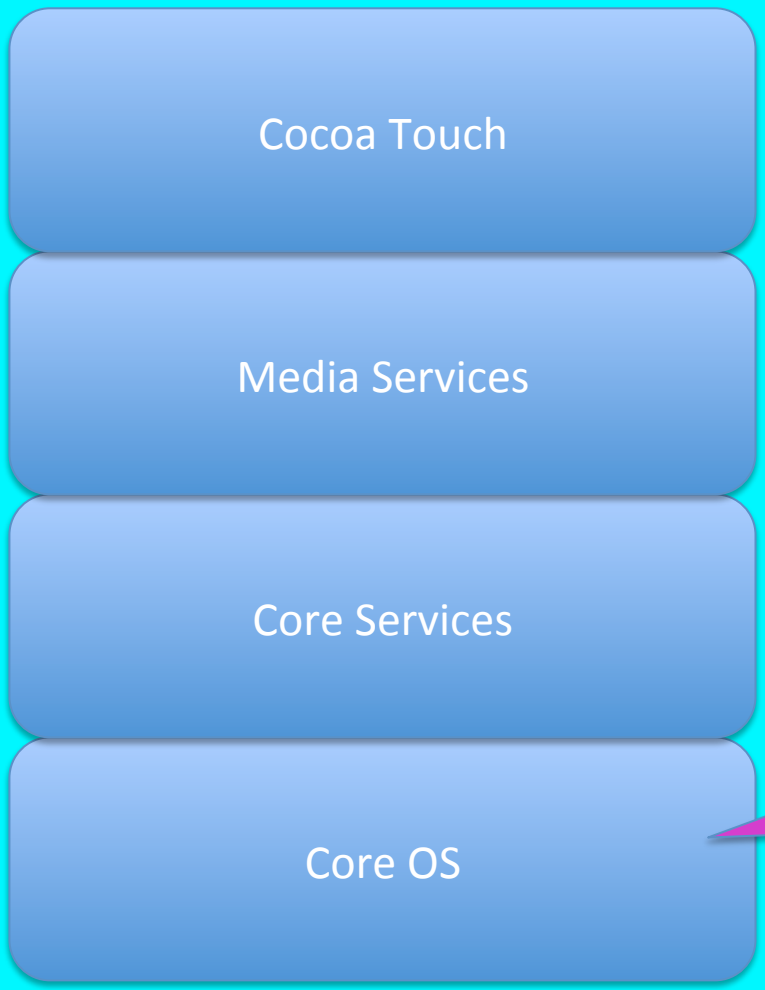
Media Services provides support for graphics, audio and video. Next week we'll start working with graphics, using this layer.

The IOS Architecture has four layers that provide amazing services.

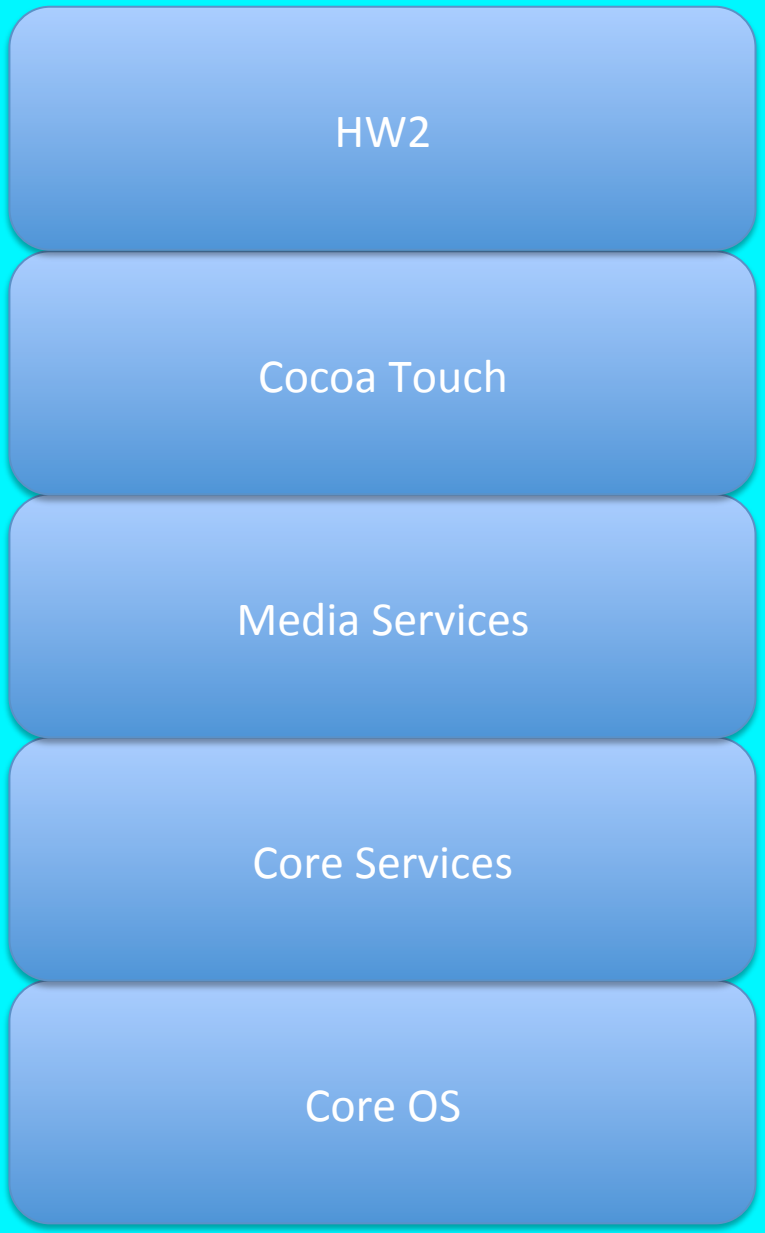


Core Services provides fundamental system services like ARC and Foundation classes (including NSObject).

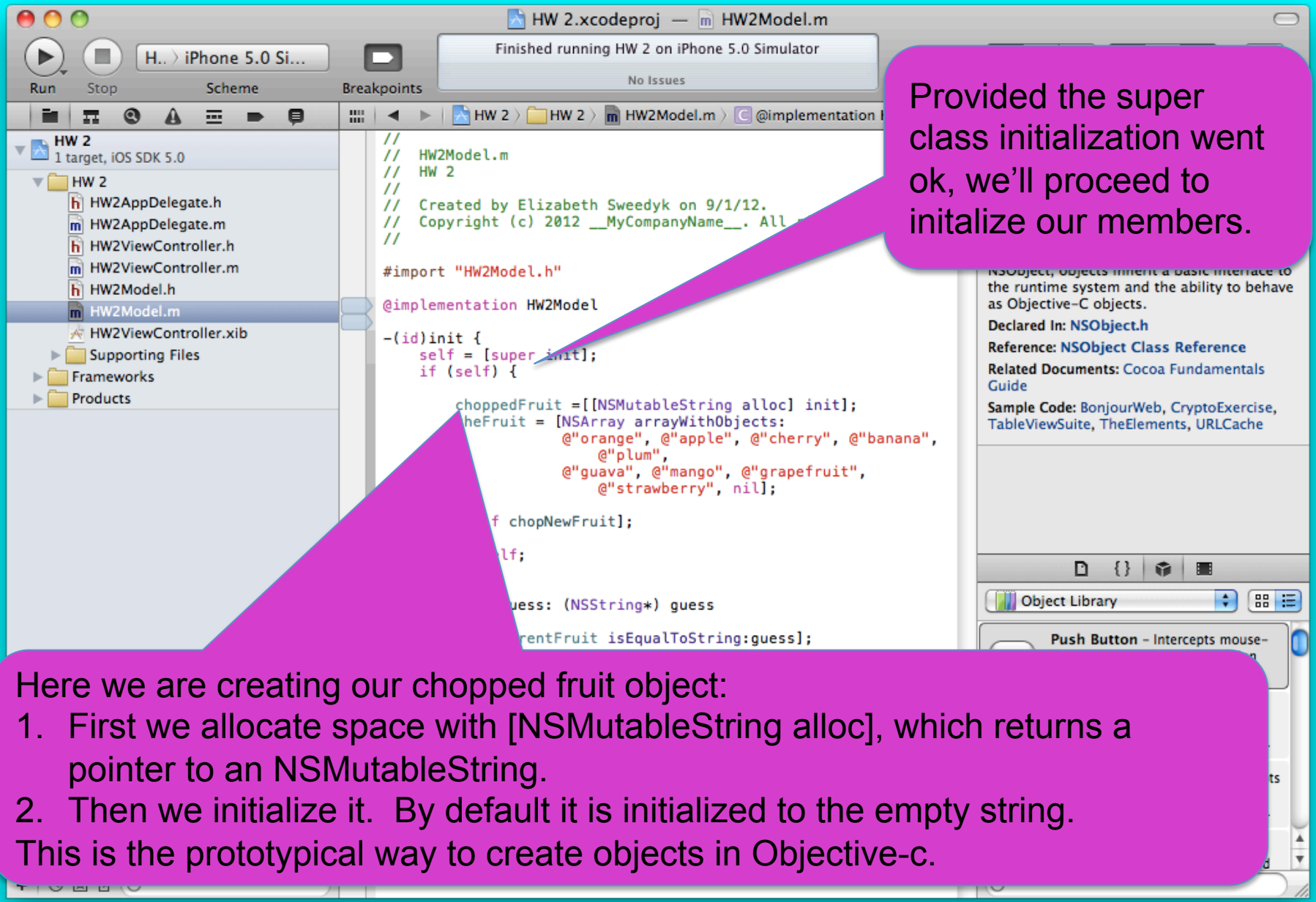
The IOS Architecture has four layers that provide amazing services.

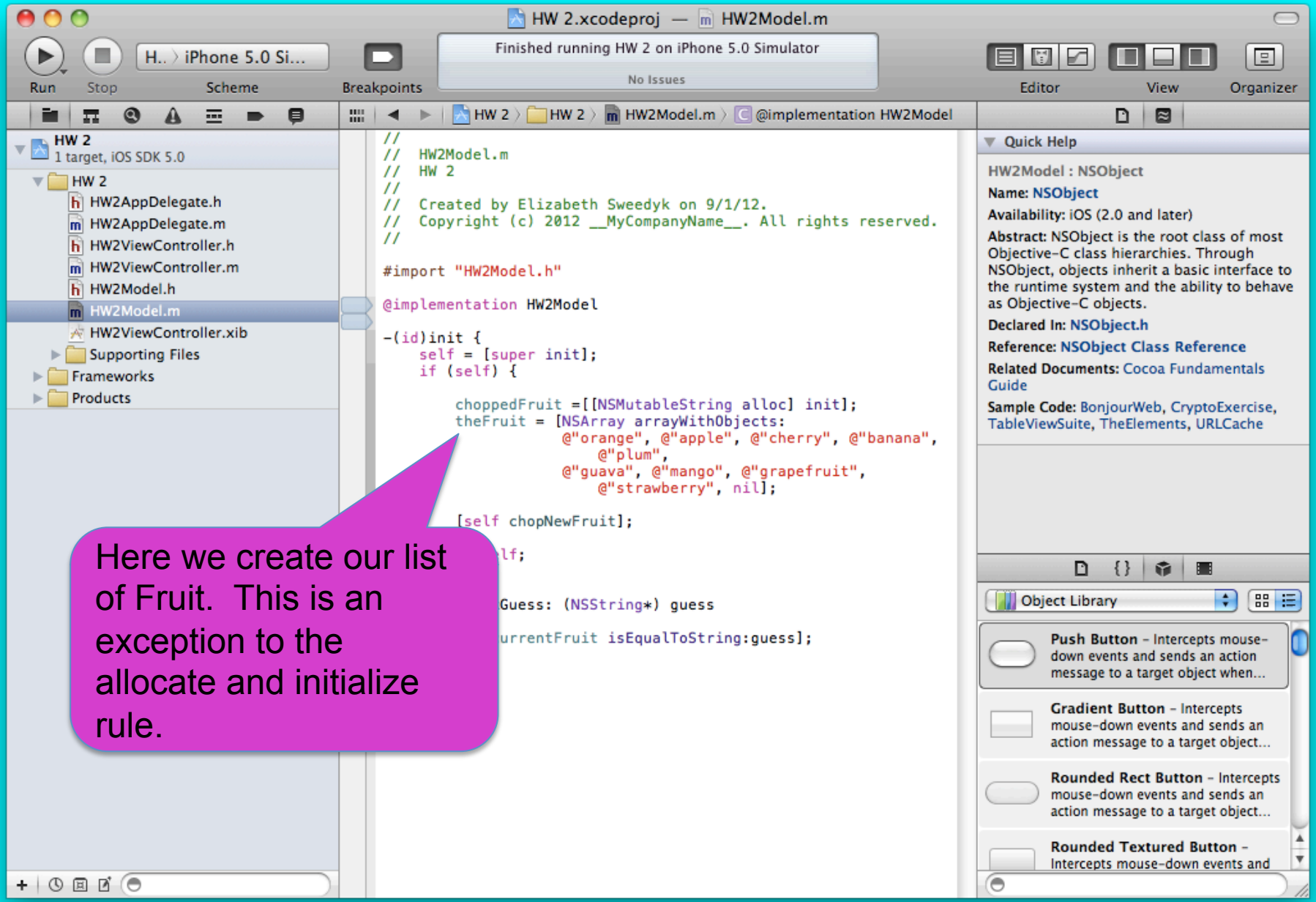


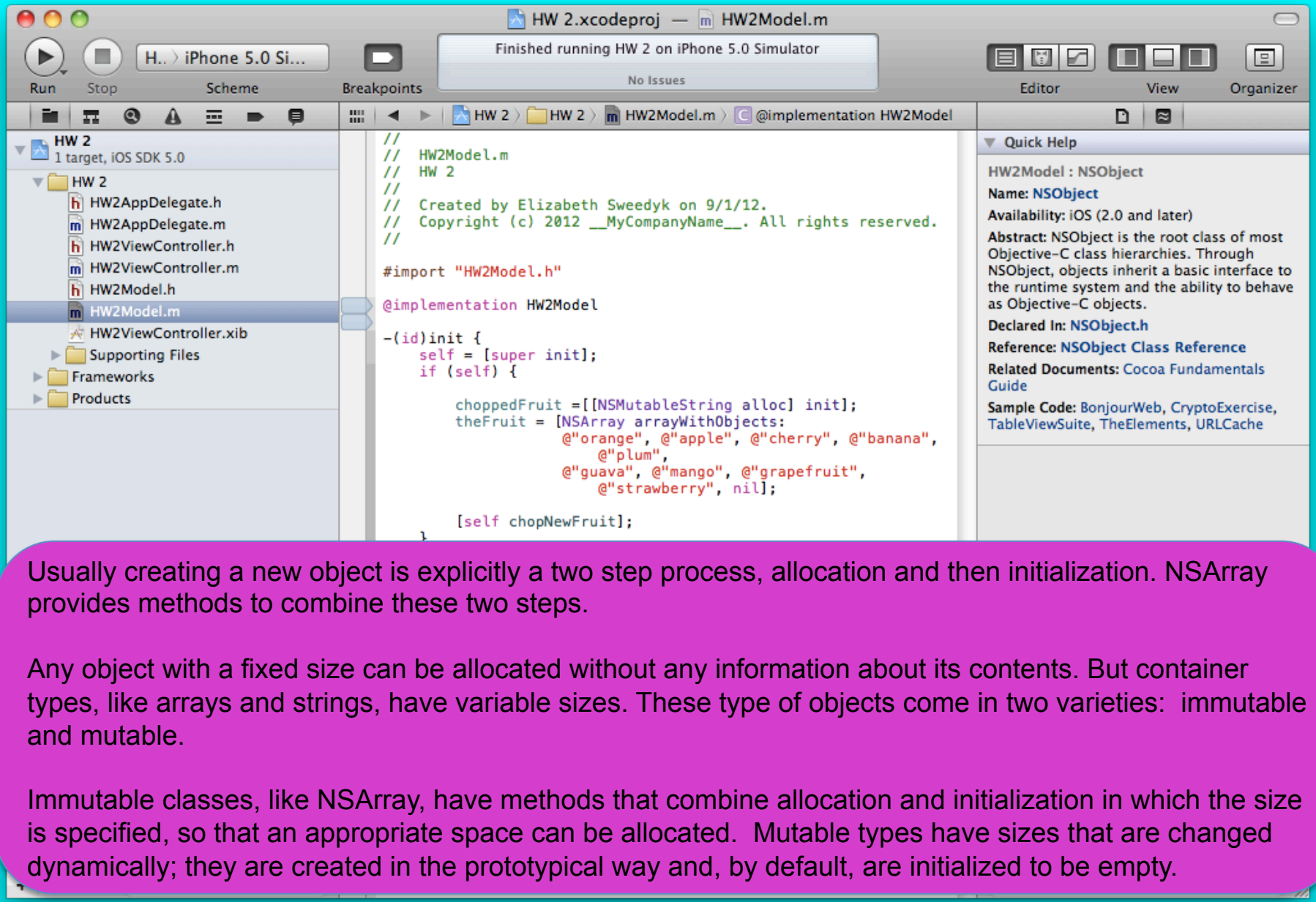
Core OS provides low level functionality that other classes build on like threading and bluetooth.



Our app sits on top but we can reach down to any layer we want; e.g. to use NSObject as a base class.


HW 2.xcodeproj — HW2Model.m
Run
Stop
Scheme
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All
//
#import "HW2Model.h"
@implementation HW2Model
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableString alloc] init];
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
Object Library
Push Button - Intercepts mouse-
Provided the super class initialization went ok, we'll proceed to initalize our members.
Here we are creating our chopped fruit object:
1. First we allocate space with [NSMutableString alloc], which returns a pointer to an NSMutableString.
2. Then we initialize it. By default it is initialized to the empty string.
This is the prototypical way to create objects in Objective-c.

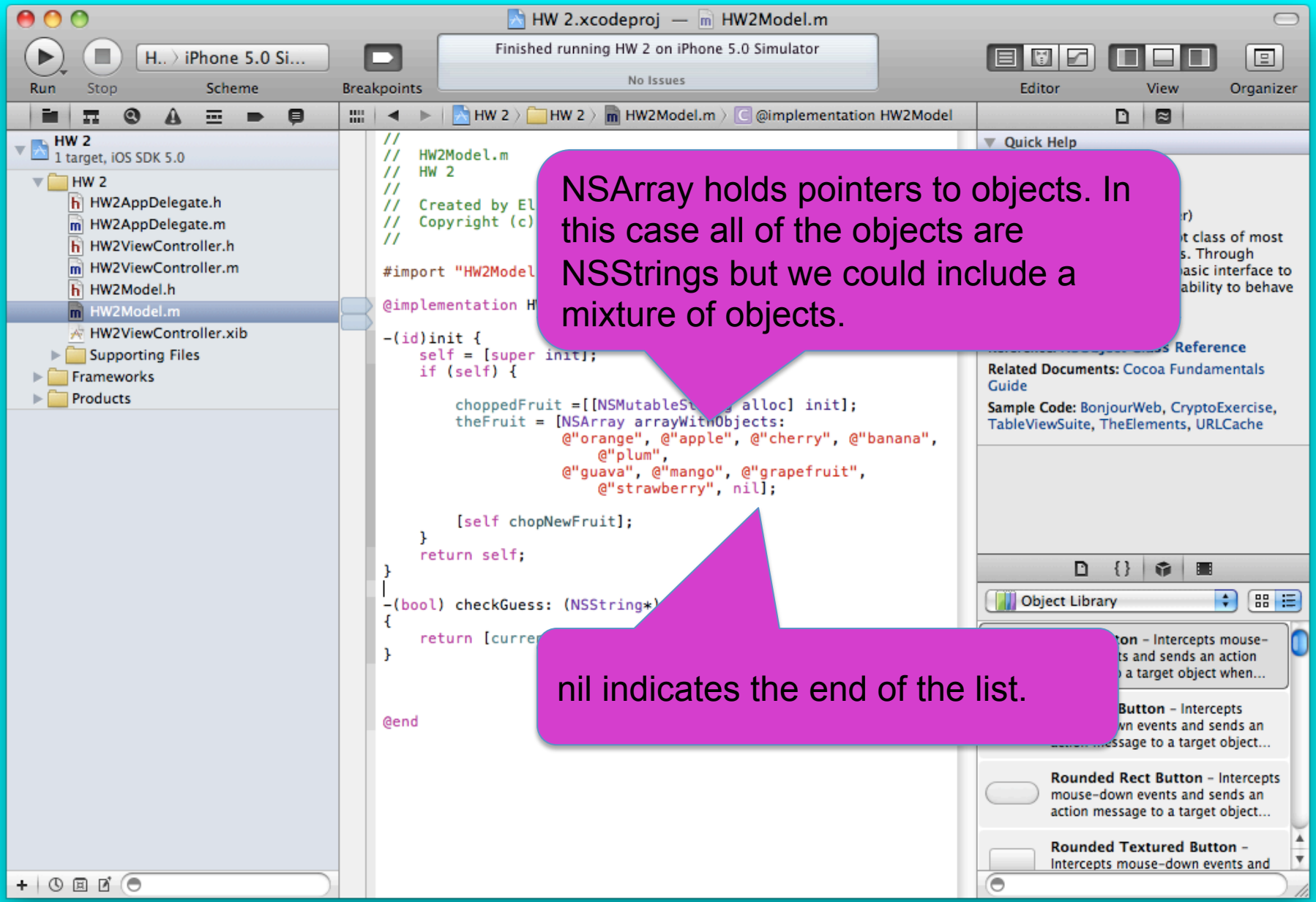

HW 2.xcodeproj — HW2Model.m
Run
Stop
H.. iPhone 5.0 Si...
Scheme
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 > HW 2 > HW2Model.m > @implementation HW2Model
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
//
#import "HW2Model.h"
@implementation HW2Model
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableString alloc] init];
theFruit = [NSArray arrayWithObjects:
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
[self chopNewFruit];
lf;
Guess: (NSString*) guess
urrentFruit isEqualToString:guess];
Here we create our list of Fruit. This is an exception to the allocate and initialize rule.
Quick Help
HW2Model : NSObject
Name: NSObject
Availability: iOS (2.0 and later)
Abstract: NSObject is the root class of most Objective-C class hierarchies. Through NSObject, objects inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.
Declared In: NSObject.h
Reference: NSObject Class Reference
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

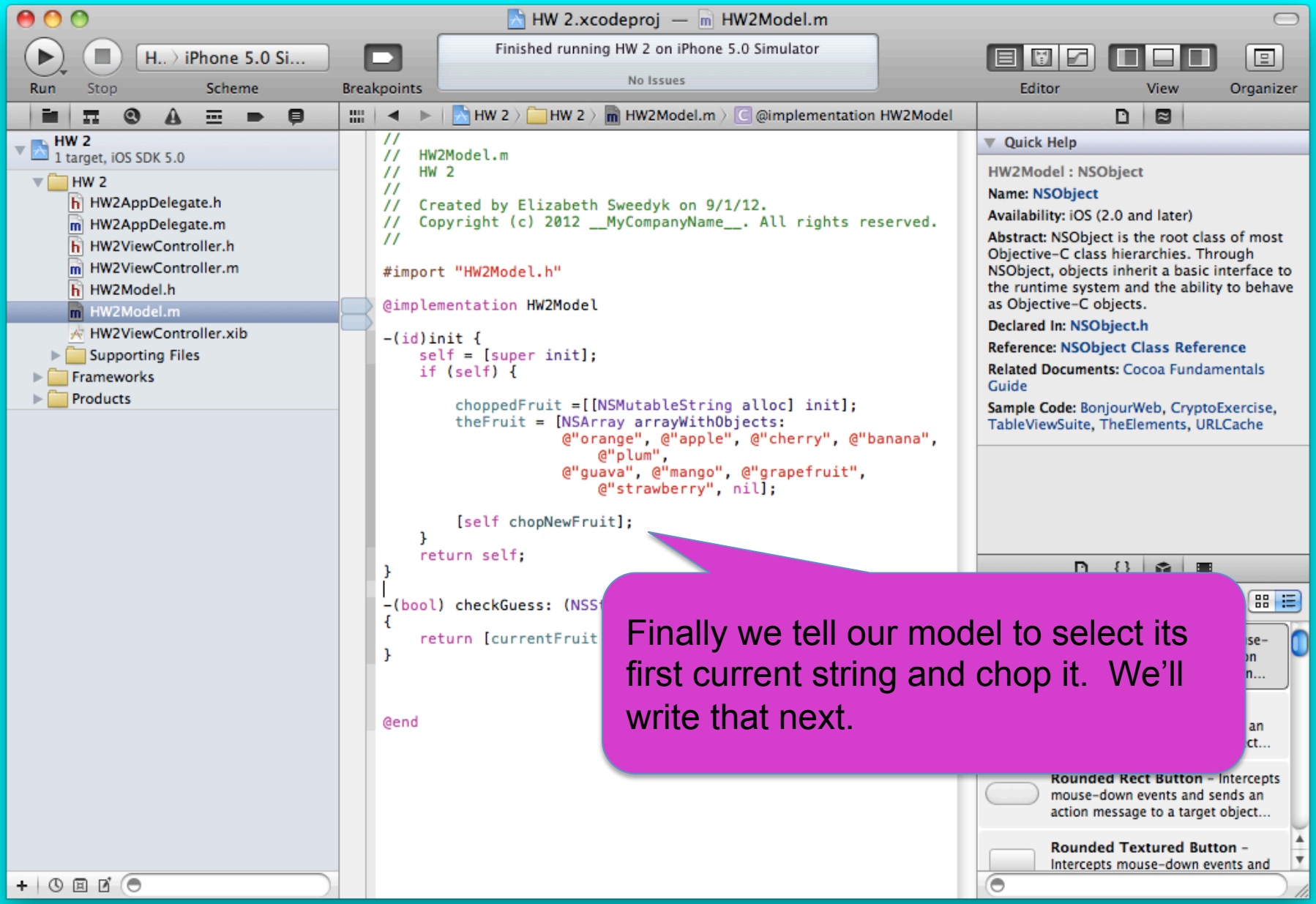


Usually creating a new object is explicitly a two step process, allocation and then initialization. NSArray provides methods to combine these two steps.

Any object with a fixed size can be allocated without any information about its contents. But container types, like arrays and strings, have variable sizes. These type of objects come in two varieties: immutable and mutable.

Immutable classes, like NSArray, have methods that combine allocation and initialization in which the size is specified, so that an appropriate space can be allocated. Mutable types have sizes that are changed dynamically; they are created in the prototypical way and, by default, are initialized to be empty.


HW 2.xcodeproj — HW2Model.m
Run
Stop
Scheme
iPhone 5.0 Si...
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 › HW 2 › HW2Model.m › @implementation HW2Model
//
// HW2Model.m
// HW 2
//
// Created by El
// Copyright (c)
//
#import "HW2Model
@implementation H
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableSt g alloc] init];
theFruit = [NSArray arrayWithObjects:
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
[self chopNewFruit];
}
return self;
}
-(bool) checkGuess: (NSString*)
{
return [curre
}
@end
NSArray holds pointers to objects. In this case all of the objects are NSStrings but we could include a mixture of objects.
nil indicates the end of the list.
Quick Help
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Object Library
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and


HW 2.xcodeproj — HW2Model.m
Run
Stop
H.. iPhone 5.0 Si...
Scheme
Breakpoints
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Editor
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 › HW 2 › HW2Model.m › @implementation HW2Model
//
// HW2Model.m
// HW 2
//
// Created by Elizabeth Sweedyk on 9/1/12.
// Copyright (c) 2012 __MyCompanyName__. All rights reserved.
//
#import "HW2Model.h"
@implementation HW2Model
-(id)init {
self = [super init];
if (self) {
choppedFruit =[[NSMutableString alloc] init];
theFruit = [NSArray arrayWithObjects:
@"orange", @"apple", @"cherry", @"banana",
@"plum",
@"guava", @"mango", @"grapefruit",
@"strawberry", nil];
[self chopNewFruit];
}
return self;
}
-(bool) checkGuess: (NSS
{
return [currentFruit
}
@end
Quick Help
HW2Model : NSObject
Name: NSObject
Availability: iOS (2.0 and later)
Abstract: NSObject is the root class of most Objective-C class hierarchies. Through NSObject, objects inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.
Declared In: NSObject.h
Reference: NSObject Class Reference
Related Documents: Cocoa Fundamentals Guide
Sample Code: BonjourWeb, CryptoExercise, TableViewSuite, TheElements, URLCache
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and
Finally we tell our model to select its first current string and chop it. We'll write that next.

This is the pseudo code for chopNewFruit:

Set currentFruit to a random element in theFruit

Copy currentFruit to choppedFruit

For (i=length(choppedFruit)-1, i>0, i--) {
    swap the character at index i with the character at a random position in [0,i]

This algorithm is called the Knuth shuffle!

Unfortunately, it is not easy to swap characters in an NSMutableString. So rather, we'll first permute the character indices then create the appropriate string. Here is the revised pseudocode.

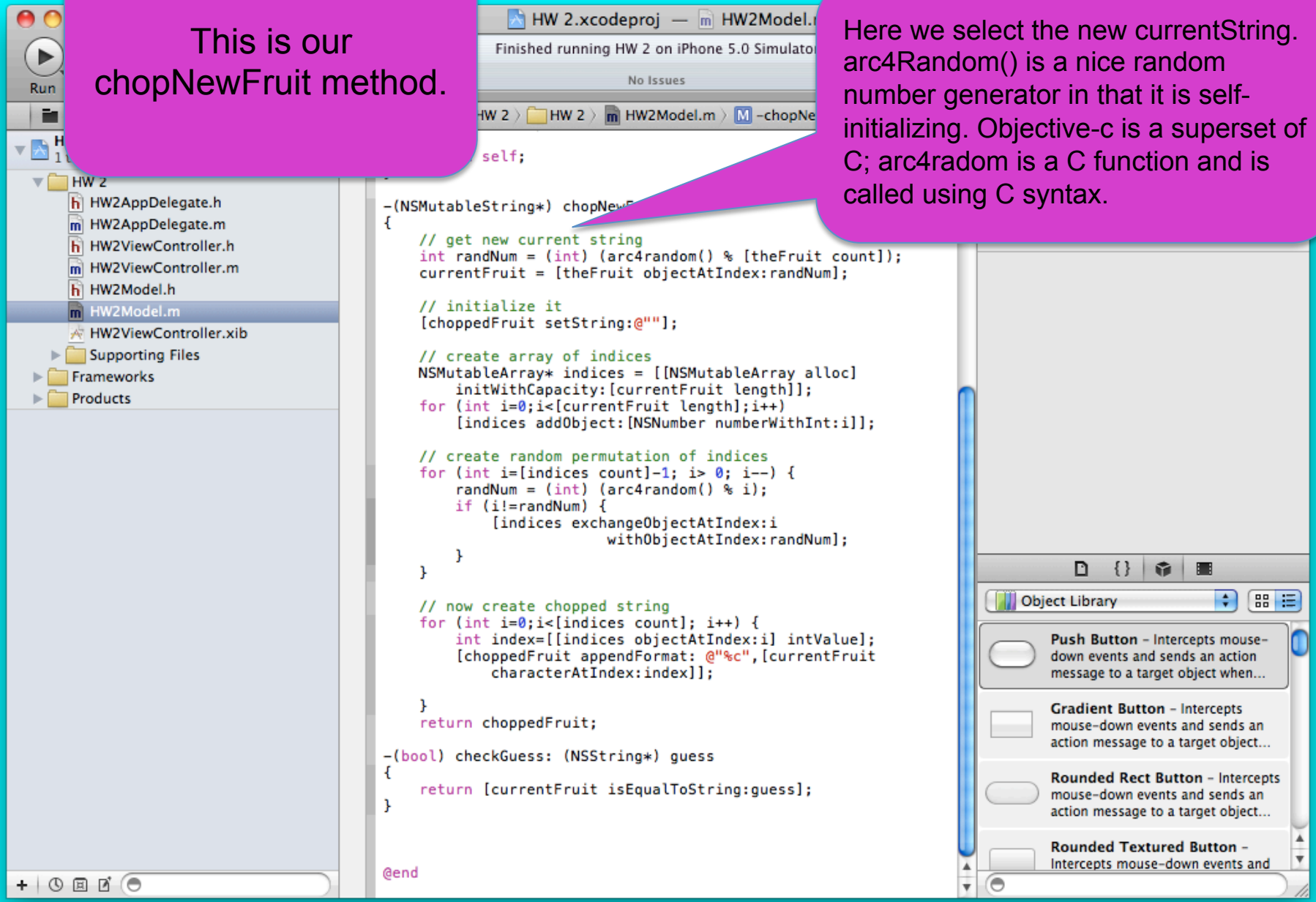
Set currentFruit to a random element in theFruit

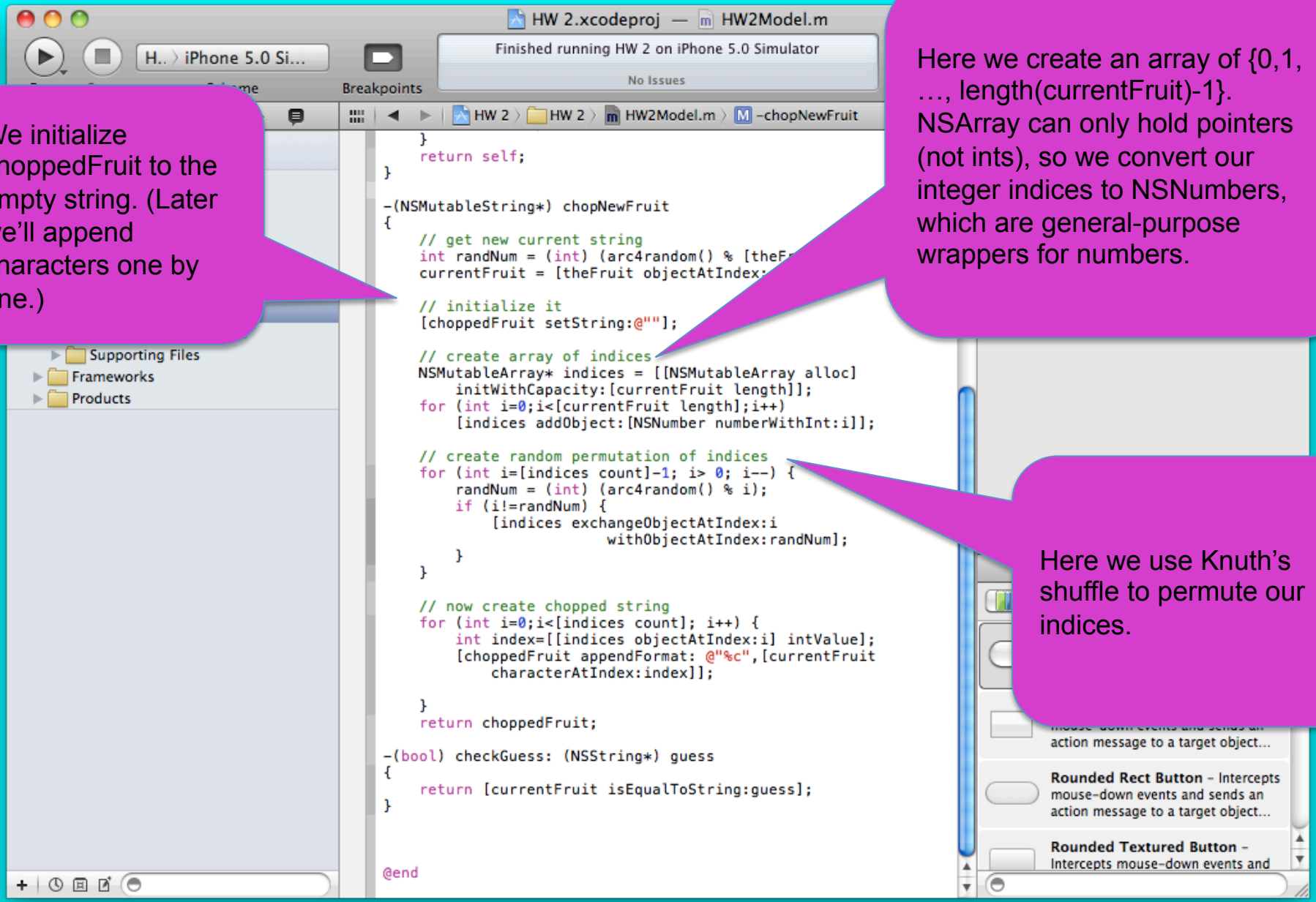
Create array of indices: 0,1,…, length(currentFruit)-1
For (i=length(currentFruit)-1, i>0, i--) {
    swap the array element at index i with the array element at a random index in [0,i]

For (i=0;i<length(currentFruit)-1)
    set i-th character of chopped fruit to the indices[i] character in currentFruit

This is our chopNewFruit method.

Here we select the new currentString. arc4Random() is a nice random number generator in that it is self-initializing. Objective-c is a superset of C; arc4radom is a C function and is called using C syntax.

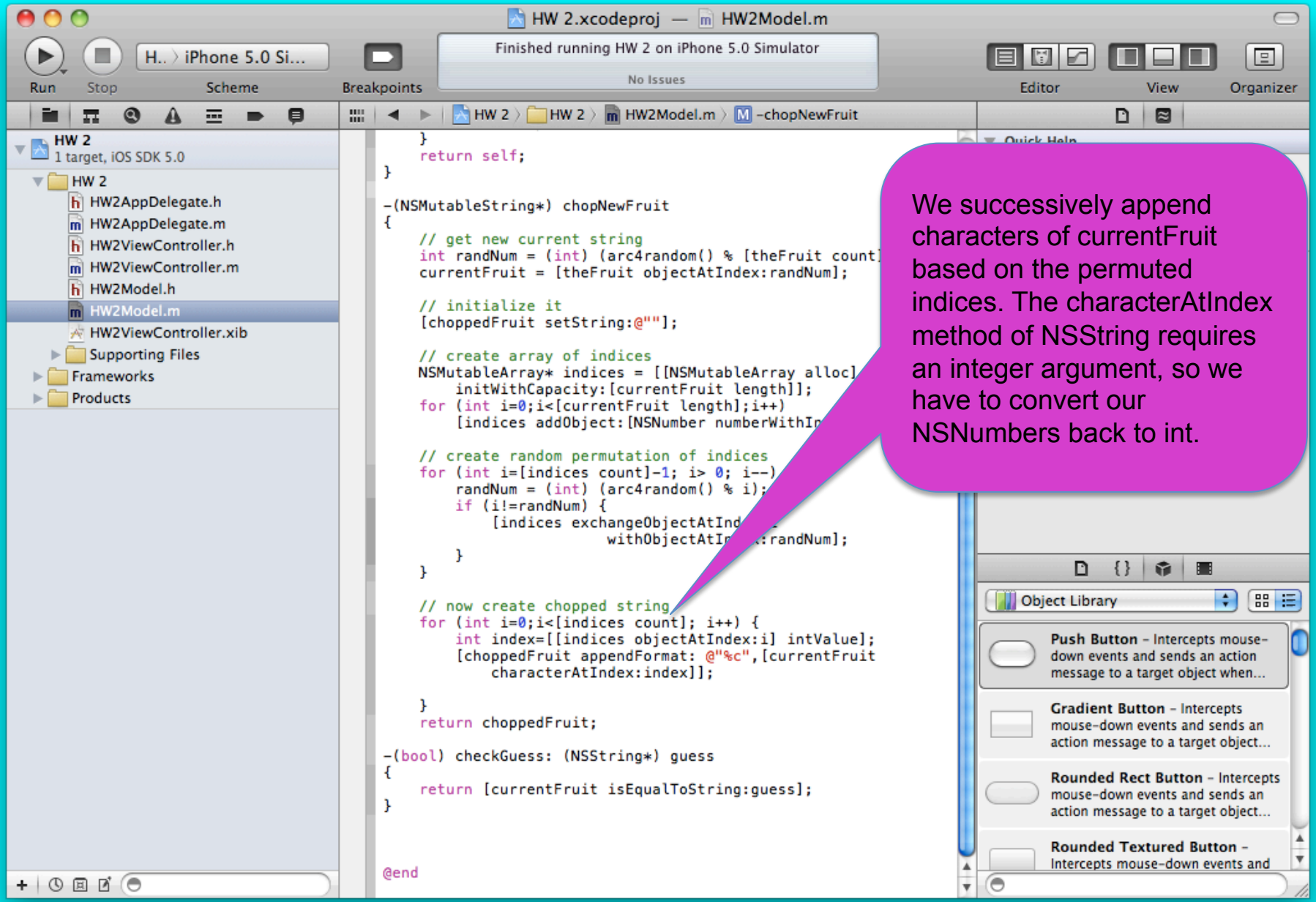




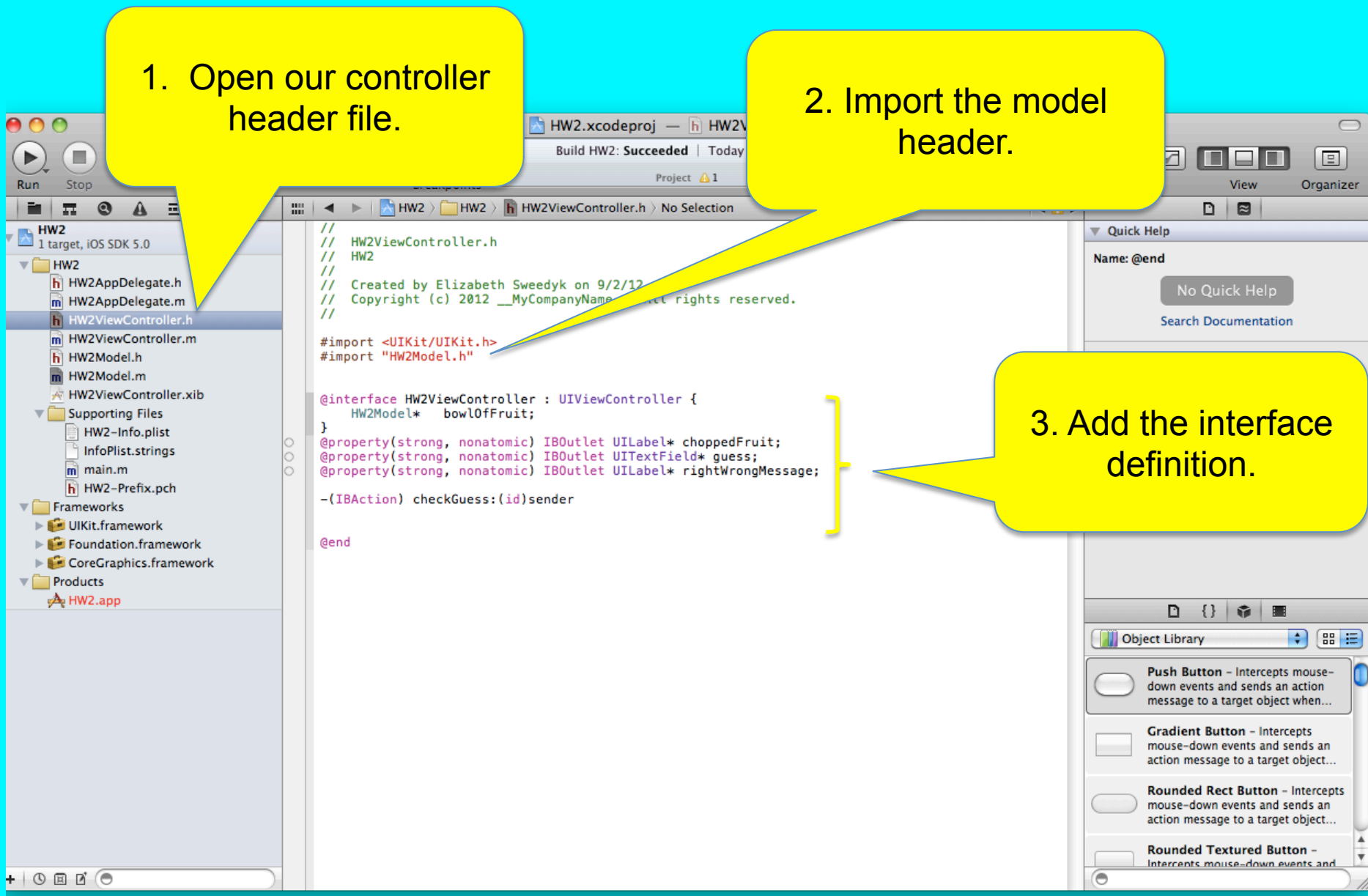
We initialize choppedFruit to the empty string. (Later we'll append characters one by one.)

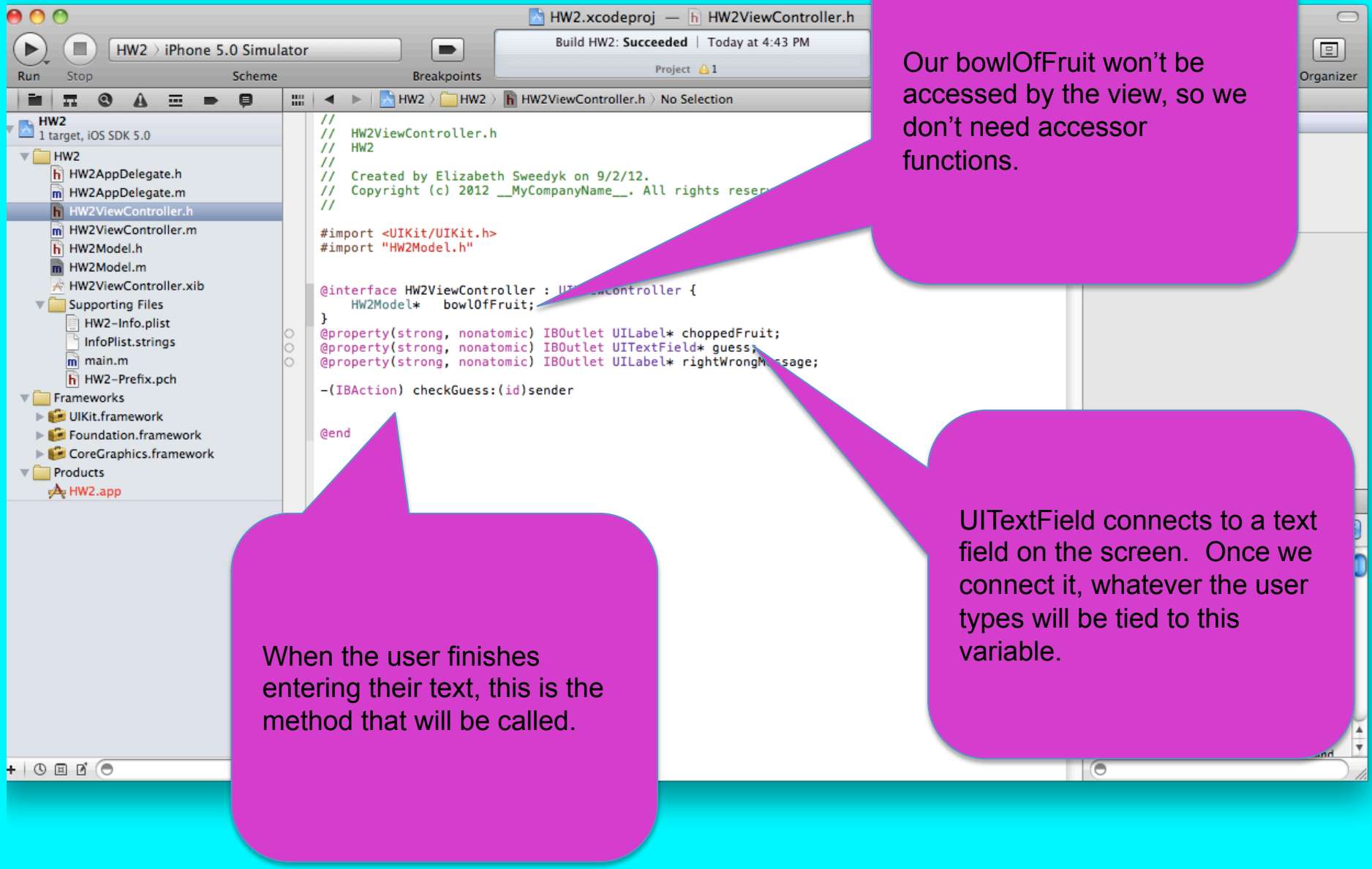
Here we create an array of {0,1, …, length(currentFruit)-1}. NSArray can only hold pointers (not ints), so we convert our integer indices to NSNumbers, which are general-purpose wrappers for numbers.

Here we use Knuth's shuffle to permute our indices.


HW 2.xcodeproj — HW2Model.m
Finished running HW 2 on iPhone 5.0 Simulator
No Issues
Run
Stop
Scheme
Breakpoints
Editor
View
Organizer
HW 2
1 target, iOS SDK 5.0
HW 2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW 2 › HW 2 › HW2Model.m › -chopNewFruit
}
return self;
}
-(NSMutableString*) chopNewFruit
{
// get new current string
int randNum = (int) (arc4random() % [theFruit count]
currentFruit = [theFruit objectAtIndex:randNum];
// initialize it
[choppedFruit setString:@""];
// create array of indices
NSMutableArray* indices = [[NSMutableArray alloc]
initWithCapacity:[currentFruit length]];
for (int i=0;i<[currentFruit length];i++)
[indices addObject:[NSNumber numberWithIn
// create random permutation of indices
for (int i=[indices count]-1; i> 0; i--)
randNum = (int) (arc4random() % i);
if (i!=randNum) {
[indices exchangeObjectAtInd
withObjectAtIn :randNum];
}
}
// now create chopped string
for (int i=0;i<[indices count]; i++) {
int index=[[indices objectAtIndex:i] intValue];
[choppedFruit appendFormat: @"%c",[currentFruit
characterAtIndex:index]];
}
return choppedFruit;
-(bool) checkGuess: (NSString*) guess
{
return [currentFruit isEqualToString:guess];
}
@end
We successively append characters of currentFruit based on the permuted indices. The characterAtIndex method of NSString requires an integer argument, so we have to convert our NSNumbers back to int.
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button – Intercepts mouse-down events and

HW2Model interface and implementation are done!!

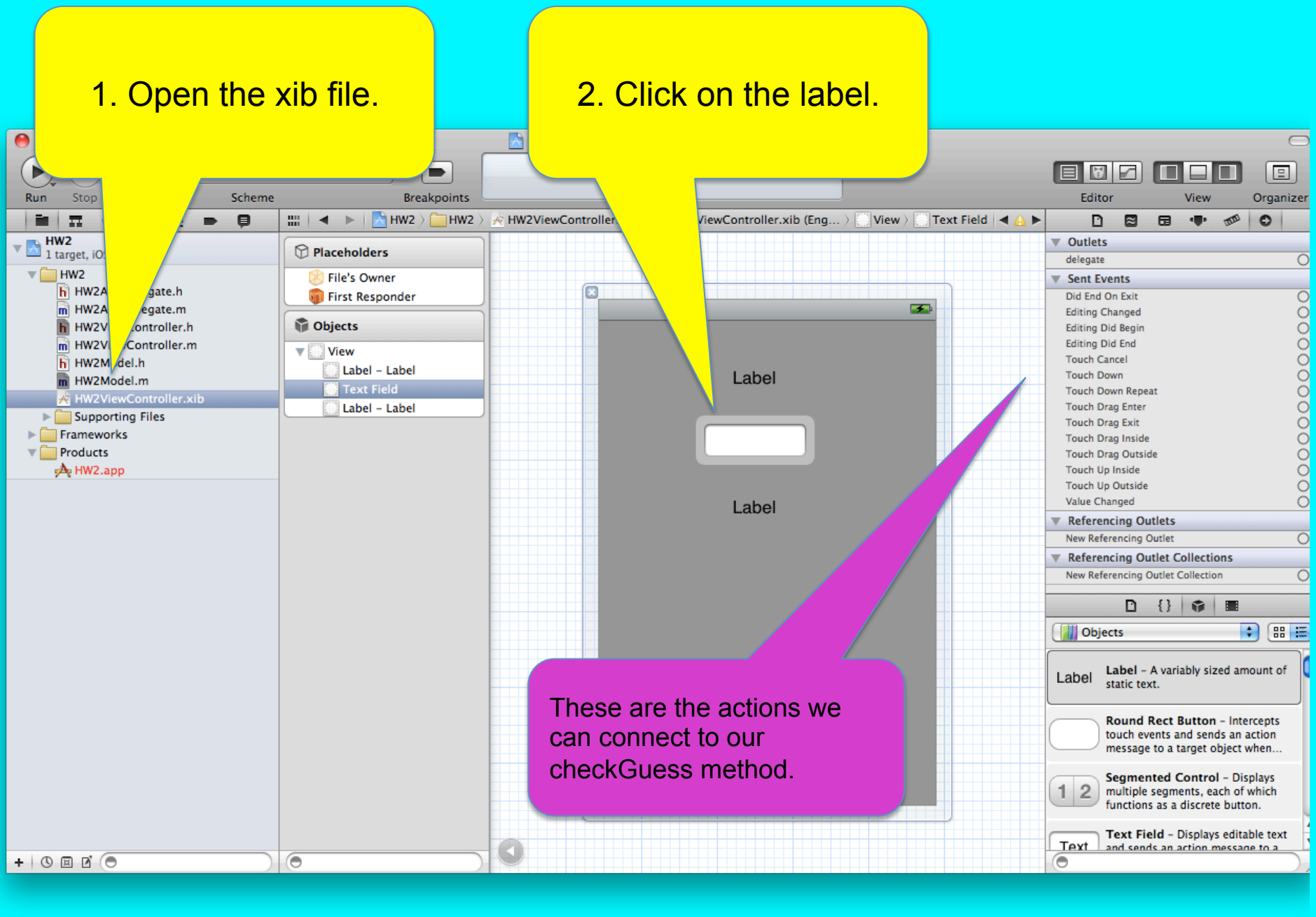

1. Open our controller header file.
2. Import the model header.
3. Add the interface definition.
HW2.xcodeproj
Build HW2: Succeeded | Today
Run
Stop
View
Organizer
HW2
1 target, iOS SDK 5.0
HW2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
HW2-Info.plist
InfoPlist.strings
main.m
HW2-Prefix.pch
Frameworks
UIKit.framework
Foundation.framework
CoreGraphics.framework
Products
HW2.app
HW2 > HW2 > HW2ViewController.h > No Selection
//
// HW2ViewController.h
// HW2
//
// Created by Elizabeth Sweedyk on 9/2/12
// Copyright (c) 2012 __MyCompanyName
rights reserved.
//
#import <UIKit/UIKit.h>
#import "HW2Model.h"
@interface HW2ViewController : UIViewController {
HW2Model* bowlOfFruit;
}
@property(strong, nonatomic) IBOutlet UILabel* choppedFruit;
@property(strong, nonatomic) IBOutlet UITextField* guess;
@property(strong, nonatomic) IBOutlet UILabel* rightWrongMessage;
-(IBAction) checkGuess:(id)sender
@end
Quick Help
Name: @end
No Quick Help
Search Documentation
Object Library
Push Button – Intercepts mouse-down events and sends an action message to a target object when...
Gradient Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...
Rounded Textured Button –

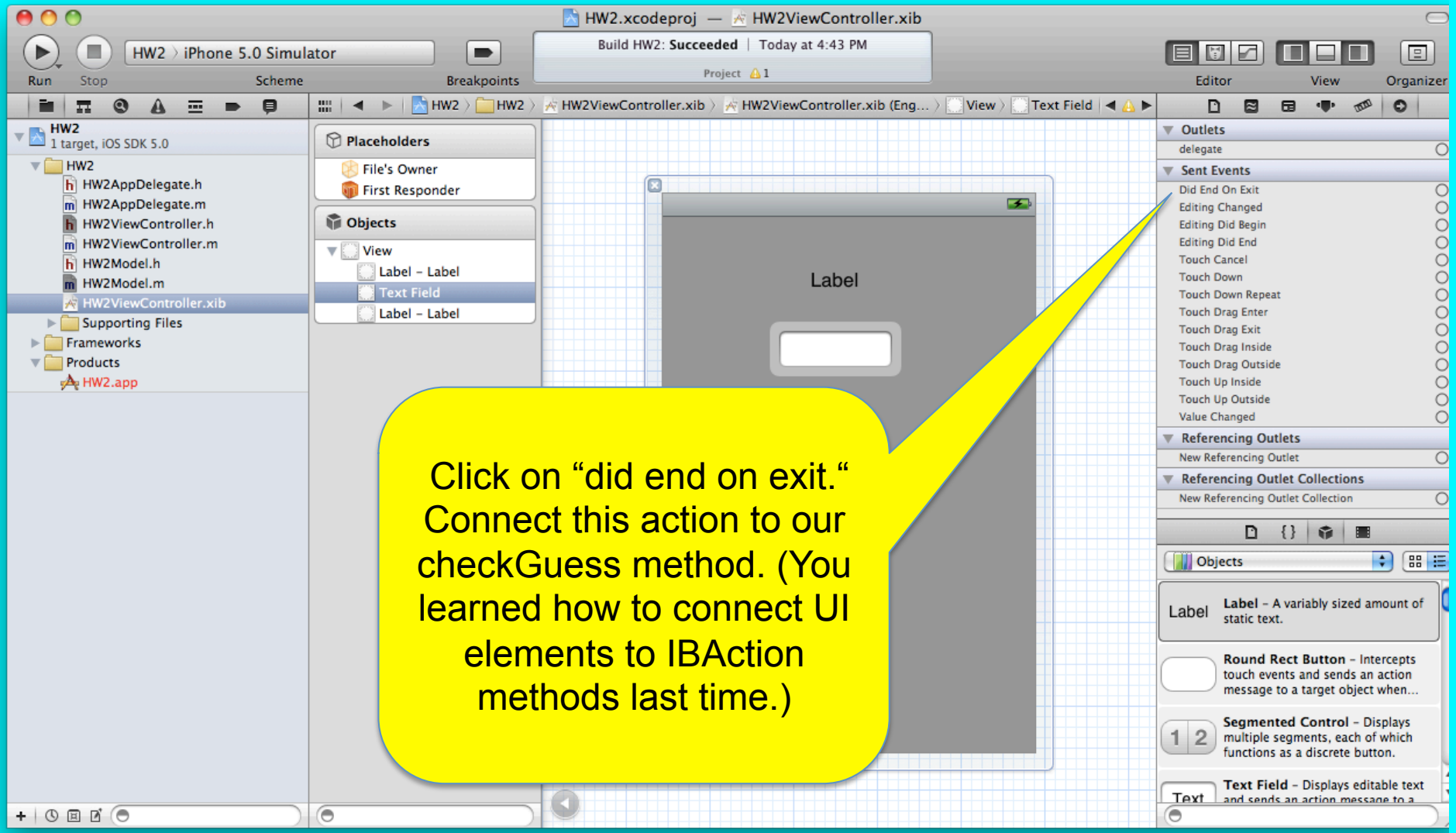


Our bowlOfFruit won't be accessed by the view, so we don't need accessor functions.

When the user finishes entering their text, this is the method that will be called.

UITextField connects to a text field on the screen. Once we connect it, whatever the user types will be tied to this variable.


1. Open the xib file.
2. Click on the label.
These are the actions we can connect to our checkGuess method.
Label
Label
Placeholders
File's Owner
First Responder
Objects
View
Label – Label
Text Field
Label – Label
HW2ViewController.xib
Outlets
delegate
Sent Events
Did End On Exit
Editing Changed
Editing Did Begin
Editing Did End
Touch Cancel
Touch Down
Touch Down Repeat
Touch Drag Enter
Touch Drag Exit
Touch Drag Inside
Touch Drag Outside
Touch Up Inside
Touch Up Outside
Value Changed
Referencing Outlets
New Referencing Outlet
Referencing Outlet Collections
New Referencing Outlet Collection
Label – A variably sized amount of static text.
Round Rect Button – Intercepts touch events and sends an action message to a target object when...
Segmented Control – Displays multiple segments, each of which functions as a discrete button.
Text Field – Displays editable text


HW2.xcodeproj — HW2ViewController.xib
Run
Stop
HW2 › iPhone 5.0 Simulator
Scheme
Breakpoints
Build HW2: Succeeded | Today at 4:43 PM
Project ⚠1
Editor
View
Organizer
HW2
1 target, iOS SDK 5.0
HW2
HW2AppDelegate.h
HW2AppDelegate.m
HW2ViewController.h
HW2ViewController.m
HW2Model.h
HW2Model.m
HW2ViewController.xib
Supporting Files
Frameworks
Products
HW2.app
Placeholders
File's Owner
First Responder
Objects
View
Label – Label
Text Field
Label – Label
HW2 › HW2 › HW2ViewController.xib › HW2ViewController.xib (Eng… › View › Text Field
Label
Outlets
delegate
Sent Events
Did End On Exit
Editing Changed
Editing Did Begin
Editing Did End
Touch Cancel
Touch Down
Touch Down Repeat
Touch Drag Enter
Touch Drag Exit
Touch Drag Inside
Touch Drag Outside
Touch Up Inside
Touch Up Outside
Value Changed
Referencing Outlets
New Referencing Outlet
Referencing Outlet Collections
New Referencing Outlet Collection
Objects
Label – A variably sized amount of static text.
Round Rect Button – Intercepts touch events and sends an action message to a target object when…
Segmented Control – Displays multiple segments, each of which functions as a discrete button.
Text Field – Displays editable text
Click on “did end on exit.“ Connect this action to our checkGuess method. (You learned how to connect UI elements to IBAction methods last time.)

You should be able to finish the app on your own.

But help is available! I will be around tomorrow until 3. Stop by or send me email if you want to meet.

Our iOS expert grutors will have grutoring hours in the LSC Saturday and Sunday 3-5 PM.