



Social Media Apps Programming

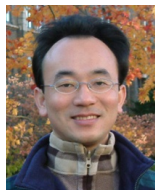
Developing iPhone / iPad Native Apps with Swift (XCode)

1041SMAP04

TLMXM1A (8687) (M2143) (Fall 2015)

(MIS MBA) (2 Credits, Elective) [Full English Course]

Wed 9,10 (16:10-18:00) B310



Min-Yuh Day, Ph.D.

Assistant Professor

Department of Information Management

Tamkang University

<http://mail.tku.edu.tw/myday>

2015-10-07



Course Schedule (1/3)

Week	Date	Subject/Topics
1	2015/09/16	Course Orientation and Introduction to Social Media and Mobile Apps Programming
2	2015/09/23	Introduction to Android / iOS Apps Programming
3	2015/09/30	Developing Android Native Apps with Java (Android Studio) (MIT App Inventor)
4	2015/10/07	Developing iPhone / iPad Native Apps with Swift (XCode)
5	2015/10/14	Mobile Apps using HTML5/CSS3/JavaScript
6	2015/10/21	jQuery Mobile

Course Schedule (2/3)

Week	Date	Subject/Topics
7	2015/10/28	Create Hybrid Apps with Phonegap
8	2015/11/04	jQuery Mobile/Phonegap
9	2015/11/11	jQuery Mobile/Phonegap
10	2015/11/18	Midterm Exam Week (Midterm Project Report)
11	2015/11/25	Case Study on Social Media Apps Programming and Marketing in Google Play and App Store
12	2015/12/02	Google Cloud Platform

Course Schedule (3/3)

Week	Date	Subject/Topics
13	2015/12/09	Google App Engine
14	2015/12/16	Google Map API
15	2015/12/23	Facebook API (Facebook JavaScript SDK) (Integrate Facebook with iOS/Android Apps)
16	2015/12/30	Twitter API
17	2016/01/06	Final Project Presentation
18	2016/01/13	Final Exam Week (Final Project Presentation)

Outline

- **Developing iPhone / iPad Native Apps with Swift 2 (Xcode 7)**
 - Mac OS X 10.8, 10.9, 10.10, 10.11
 - Xcode 6, Xcode 7
 - iOS 8, iOS 9
- **Building Your First iOS App with Xcode 7**

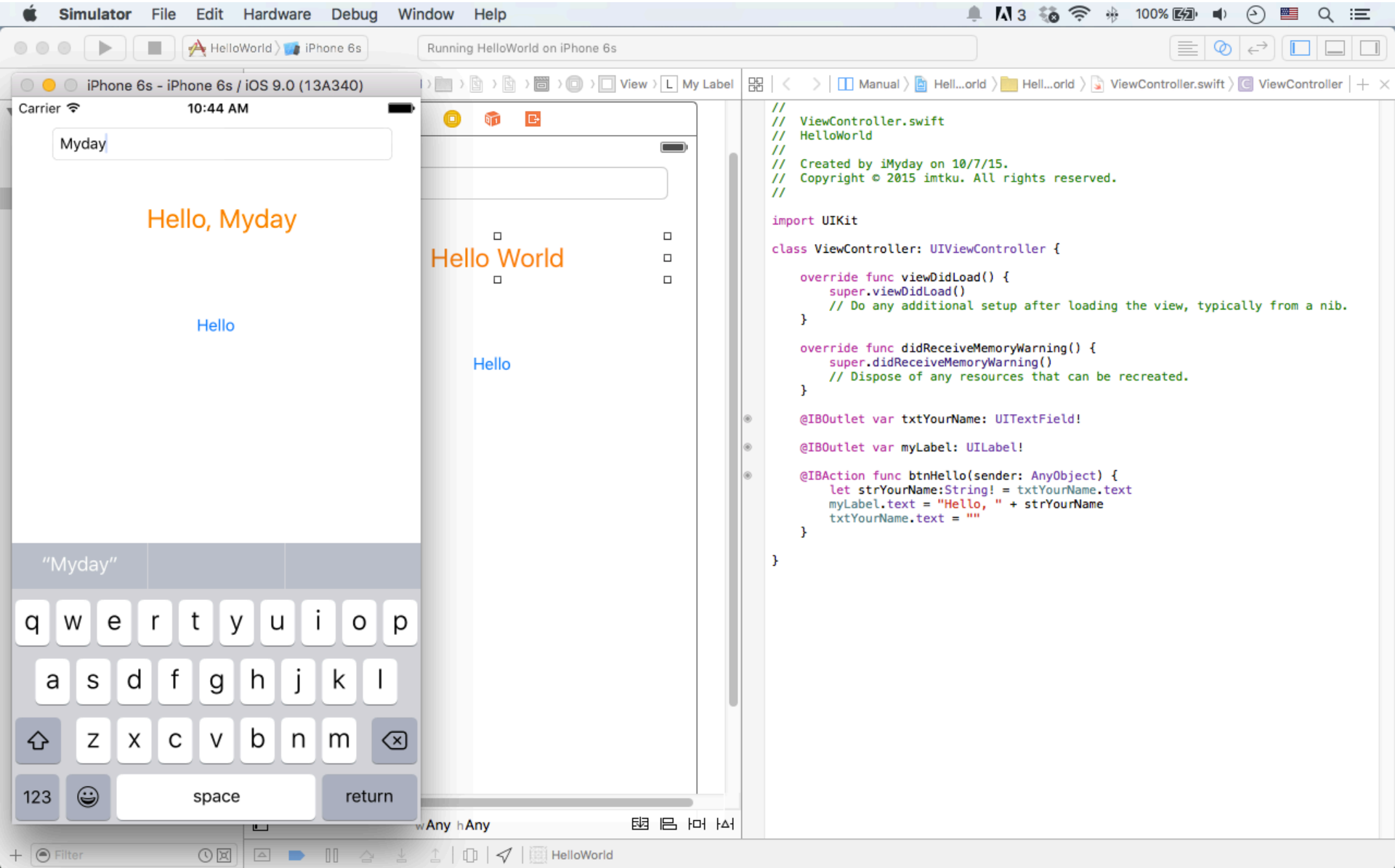


Xcode 7



Swift 2

Building Your First iOS App with Xcode 7



Android /iOS Apps Programming

Native Apps

Hybrid Apps

Mobile Web Apps



App Development Comparison

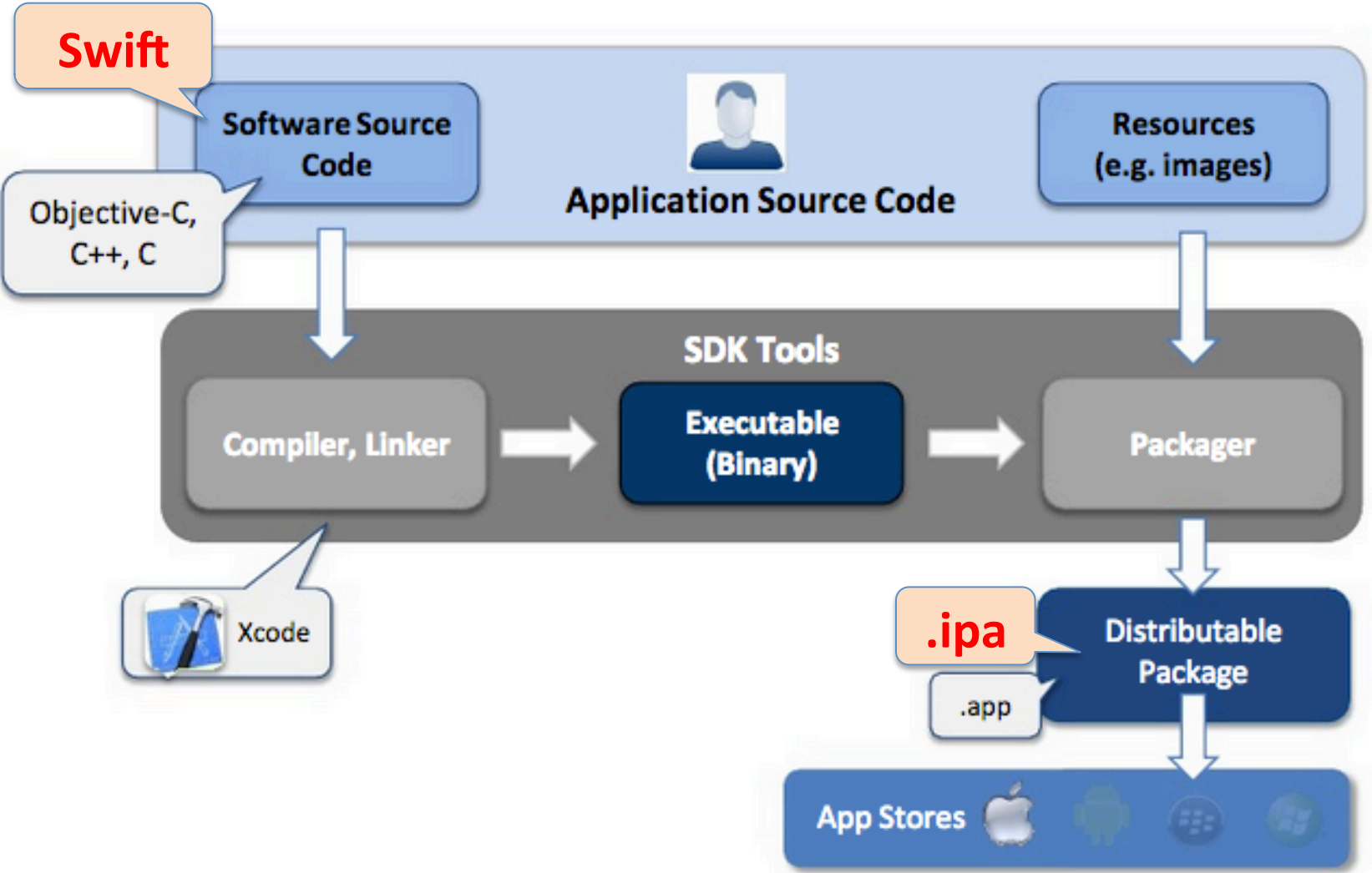
Device Access	Speed	Development Cost	App Store	Approval Process
---------------	-------	------------------	-----------	------------------

Native Apps	Full	Very Fast	Expensive	Available	Mandatory
--------------------	------	-----------	-----------	-----------	-----------

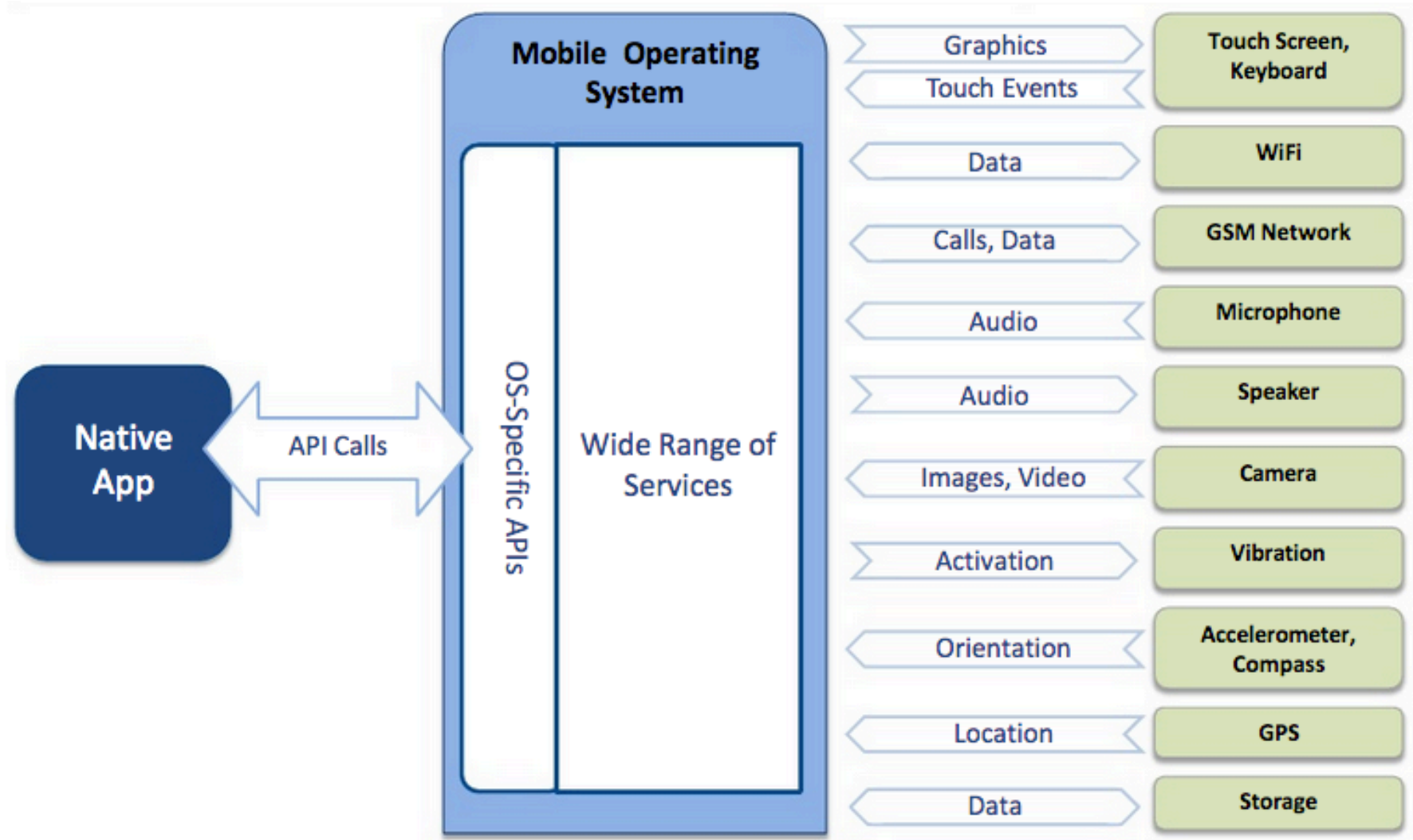
Hybrid Apps	Full	Native Speed as Necessary	Reasonable	Available	Low Overhead
--------------------	------	---------------------------	------------	-----------	--------------

Web Apps	Partial	Fast	Reasonable	Not Available	None
-----------------	---------	------	------------	---------------	------

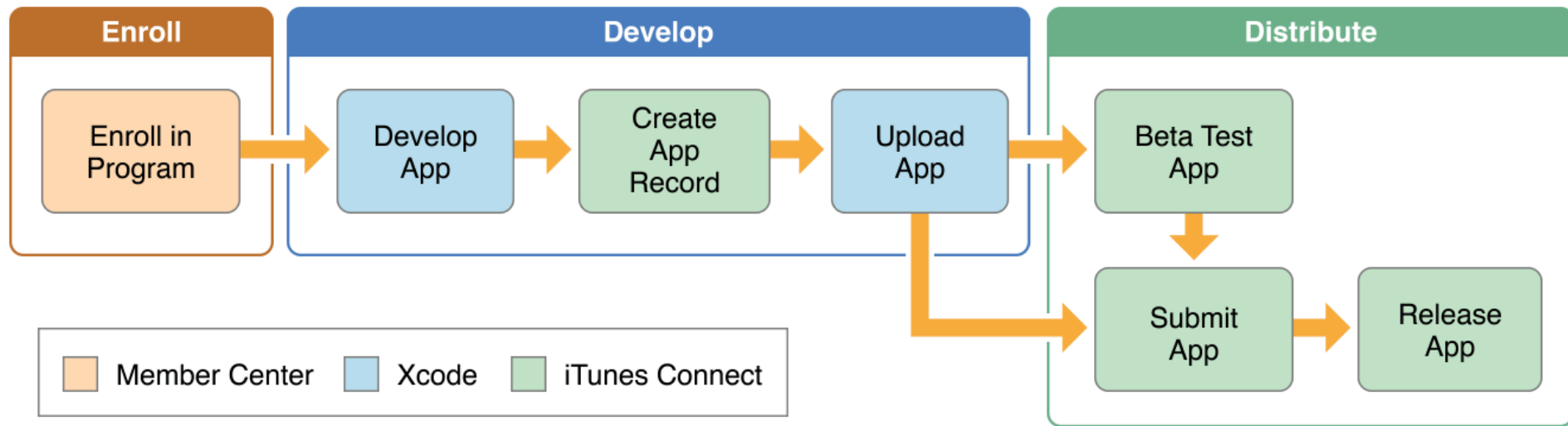
iOS - Native App Development



Native App – Interaction with Mobile Device



Apple App Distribution Workflows



developer.apple.com

 Developer

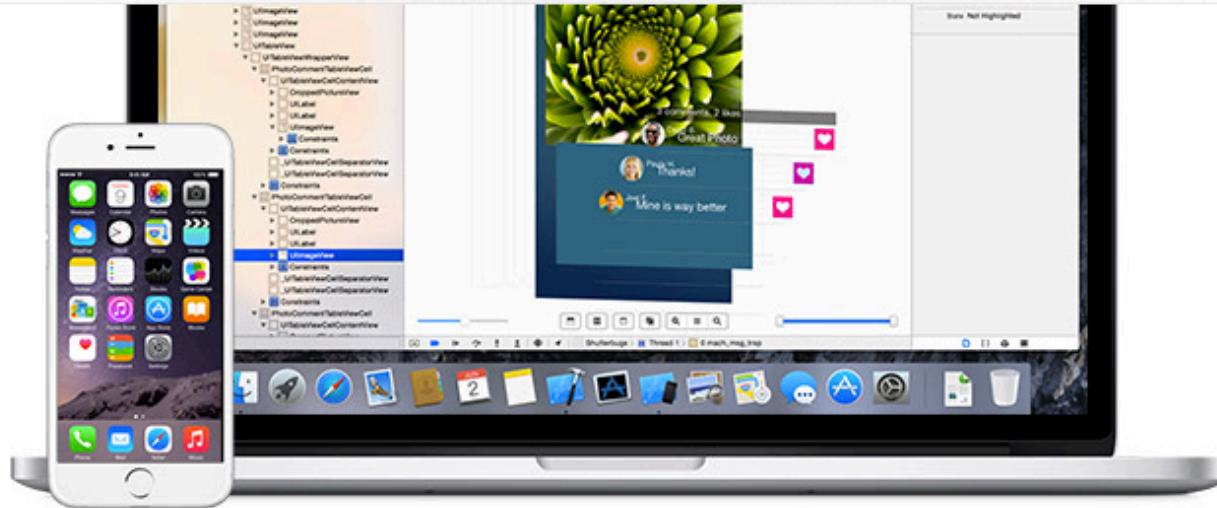
Technologies

Resources

Programs

Support

Member Center



See what's new for developers.

Learn about all the new technologies and powerful capabilities available in iOS 8, OS X Yosemite, and the new programming language, Swift, available in Xcode 6.

developer.apple.com

Download the latest development tools and SDKs.



iOS Apps



Mac Apps



Xcode

developer.apple.com

Become a member.

Enroll in an Apple Developer Program and get everything you need to develop and distribute your apps.

[Learn more >](#)



Apress
For professionals by professionals

Your Account | Your Cart | Sign In

Hot Topics: .NET | iPhone | Android | Web Development | Show All Categories

Search apress.com... Go

Home | Alpha Program | Springer | Source Code | About Apress | Support | Interact with Apress

Twitter | Facebook

Home » iPhone and iPad Apps for Absolute Beginners

iPhone and iPad Apps for Absolute Beginners

4th Edition

By **Rory Lewis, Laurence Moroney**

This update of an Apress bestseller teaches you how to create your first iOS 7 app to run on iPhone or iPad, using plain English and practical examples. It cuts through the jargon that surrounds iPhone and iPad app development with simple, step-by-step instructions to get you started.

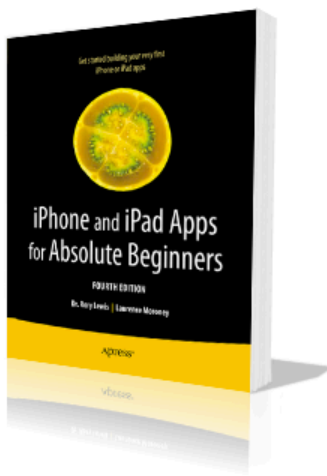
Full Description

ISBN13: 978-1-4302-6361-6


400 Pages

User Level: Beginner

Publishing October 23, 2013, but available now as part of the Alpha Program



Your Cart



You don't have any books in your shopping cart yet!

Deal of the Day




Pro SharePoint 2013 Branding and Responsive Web Development

Pro SharePoint 2013 Branding and Responsive Web Development

Start Developing iOS Apps Today

The screenshot shows the Apple Developer Library page for the document 'Start Developing iOS Apps Today'. The page is divided into several sections:

- Introduction**
 - **Setup**
 - Tutorial: Basics
- Structuring an App**
 - App Development Process
 - Designing a User Interface
 - Defining the Interaction
 - Tutorial: Storyboards
- Implementing an App**
 - Incorporating the Data
 - Using Design Patterns
 - Working with Foundation
 - Writing a Custom Class
 - Tutorial: Add Data
- Next Steps**
 - iOS Technologies
 - Finding Information
 - Where to Go from Here

The main content area is titled **Setup** and contains the following text:

Start Developing iOS Apps Today provides the perfect starting point for iOS development. On your Mac, you can create iOS apps that run on iPad, iPhone, and iPod touch. View this guide's four short modules as a gentle introduction to building your first app—including the tools you need and the major concepts and best practices that will ease your path.

The content includes a storyboard diagram showing a navigation controller with three view controllers: 'New Details', 'Map', and 'Coffees'. Below the storyboard is a code snippet for the `BRWDetailsViewController` class, which implements `UITableViewDataSource` and `UITableViewDelegate` protocols. The code shows how to fetch coffee data from a remote source and display it in a table view.

The first three modules each end with a tutorial, where you'll implement what you've learned. At the end of the last tutorial, you'll have created a simple to-do list app.

Provide Feedback

Start Developing iOS Apps Today

```
#import <CoreData/CoreData.h>
#import "BrewSessionsViewController.h"
#import "BrewSessionDetailsViewController.h"
#import "BrewSessionManagedObject.h"
#import "CoffeesViewController.h"
#import "DailyBuzzAppDelegate.h"
#import "CoffeeManagedObject.h"

@interface BrewSessionsViewController () <NSFetchingResultsControllerDelegate, CoffeesViewControll
@property (strong, nonatomic) NSFetchingResultsController *fetchResultsController;
@end

@implementation BrewSessionsViewController
@synthesize fetchResultsController = _fetchResultsController;

- (NSFetchingResultsController *)fetchResultsController {
    if (nil == _fetchResultsController) {
        NSManagedObjectContext *managedObjectContext = [(DailyBuzzAppDelegate *)[[UIApplication
        managedObjectContext]];

        NSFetchRequest *fetchRequest = [[NSFetchRequest alloc] init];
        // Edit the entity name as appropriate.
        NSEntityDescription *entity = [NSEntityDescription entityForName:@"BrewSession"
        managedObjectContext];
        [fetchRequest setEntity:entity];

        // Set the batch size to a suitable number.
        [fetchRequest setFetchBatchSize:20];

        // Edit the sort key as appropriate.
        NSSortDescriptor *sortDescriptor = [[NSSortDescriptor alloc] initWithKey:@"date"
        NSArray *sortDescriptors = [[NSArray alloc] initWithObjects:sortDescriptor, nil];
        [fetchRequest setSortDescriptors:sortDescriptors];

        NSFetchingResultsController *aFetchingResultsController = [[NSFetchingResultsController
        fetchRequest managedObjectContext:managedObjectContext sectionNameKeyPath:nil
        aFetchingResultsController.delegate = self;
        self.fetchResultsController = aFetchingResultsController;

        NSError *error = nil;
        if (![self.fetchResultsController performFetch:&error]) {
            NSLog(@"Unresolved error %@, %@", error, [error userInfo]);
            abort();
        }
    }
    return _fetchResultsController;
}
```

Carrier 18:52

Cancel Coffees

Brazilian

- Cerrado**
Bright, with a light cocoa touch.
- Sul de Minas**
Hints of chocolate & nuts with minor acidity.

Costa Rican

- Tarrazu**
Mid-body, with a strong nuttiness.
- Tres Rios**
Full body followed by a surge of citrus.

Rwandan

- Musasa**
Low-body, with an overwhelming

To develop iOS apps, you need:

- A **Mac computer** running OS X 10.8 or later
 - Mac OS X 10.8 (Mountain Lion)
 - Mac OS X 10.9 (Mavericks)
 - Mac OS X 10.10 (Yosemite)
 - Mac OS X 10.11 (El Capitan)
- Xcode
 - Xcode 6
 - Xcode 7
- iOS SDK



Mac

iPad

iPhone

Watch

TV

Music

Support



MacBook



MacBook Air



MacBook Pro



iMac



iMac Retina 5K display



Mac Pro



Mac mini



Accessories



OS X Yosemite



Compare



MacBook

Light. Years ahead.

[Learn more >](#) [Buy >](#)



Store

Mac

iPhone

Watch

iPad

iPod

iTunes

Support



MacBook Air



MacBook Pro



Mac mini



iMac



Mac Pro



OS X Mavericks

Mac

Apps

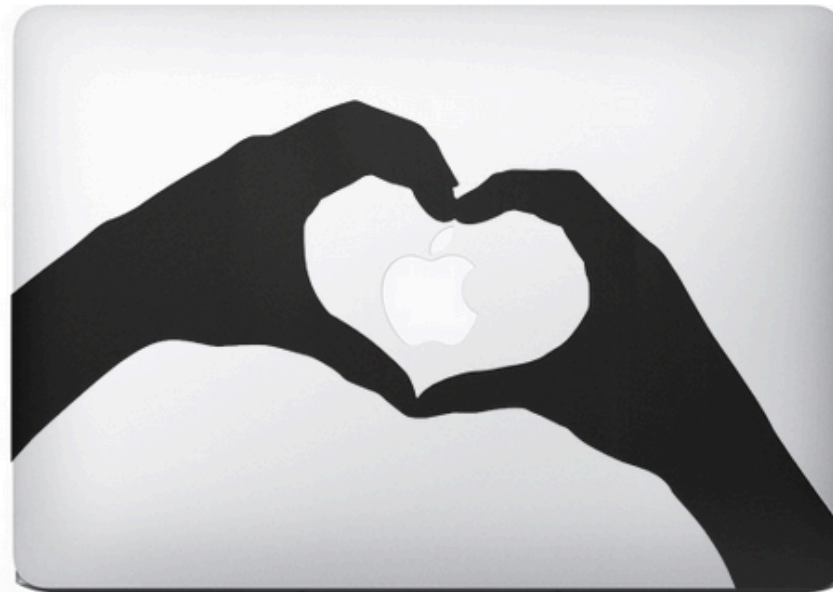
Pro Apps

Accessories

Server

The notebook people love.

MacBook Air



MacBook and MacBook Air



MacBook
from \$1299

- 12-inch (diagonal) LED-backlit Retina display
- 1.1GHz, 1.2GHz, or 1.3GHz dual-core Intel Core M processor
Turbo Boost up to 2.9GHz
- Up to 9 hours battery life¹
- Up to 512GB flash storage²
- 2.03 pounds³
- Available in gold, silver, and space gray
- The world's most energy-efficient notebook⁴



MacBook Air 11-inch
from \$899

- 11.6-inch (diagonal) LED-backlit display
- 1.6GHz dual-core Intel Core i5 or 2.2GHz dual-core Intel Core i7 processor
Turbo Boost up to 3.2GHz
- Up to 9 hours battery life¹
- Up to 512GB flash storage²
- 2.38 pounds³



MacBook Air 13-inch
from \$999

- 13.3-inch (diagonal) LED-backlit display
- 1.6GHz dual-core Intel Core i5 or 2.2GHz dual-core Intel Core i7 processor
Turbo Boost up to 3.2GHz
- Up to 12 hours battery life¹
- Up to 512GB flash storage²
- 2.96 pounds³

MacBook Pro



MacBook Pro 13-inch
from \$1099

- 13.3-inch (diagonal) LED-backlit display
- 2.5GHz dual-core Intel Core i5 or 2.9GHz dual-core Intel Core i7 processor
Turbo Boost up to 3.6GHz
- Up to 7 hours battery life¹
- Up to 1TB 5400-rpm hard drive; or up to 512GB solid-state drive²
- 4.5 pounds³



MacBook Pro 13-inch
with Retina display
from \$1299

- 13.3-inch (diagonal) LED-backlit Retina display
- 2.7GHz or 2.9GHz dual-core Intel Core i5 or 3.1GHz dual-core Intel Core i7 processor
Turbo Boost up to 3.4GHz
- Up to 10 hours battery life¹
- Up to 1TB flash storage²
- 3.48 pounds³



MacBook Pro 15-inch
with Retina display
from \$1999

- 15.4-inch (diagonal) LED-backlit Retina display
- 2.2GHz, 2.5GHz, or 2.8GHz quad-core Intel Core i7 processor
Turbo Boost up to 4.0GHz
- Up to 9 hours battery life¹
- Up to 1TB flash storage²
- 4.49 pounds³

iMac



iMac 21.5-inch
from \$1099

- 21.5-inch (diagonal) LED-backlit display
- 1.4GHz dual-core, 2.7GHz quad-core, or 2.9GHz quad-core Intel Core i5 or 3.1GHz quad-core Intel Core i7 processor
Turbo Boost up to 3.9GHz
- 500GB or 1TB 5400-rpm hard drive; 1TB Fusion Drive; or up to 512GB flash storage (SSD)²
- Apple Wireless Keyboard and Magic Mouse or Magic Trackpad



iMac 27-inch
from \$1799

- 27-inch (diagonal) LED-backlit display
- 3.2GHz quad-core Intel Core i5 processor
Turbo Boost up to 3.6GHz
- 1TB or 3TB 7200-rpm hard drive; 1TB or 3TB Fusion Drive; or up to 1TB flash storage (SSD)²
- Apple Wireless Keyboard and Magic Mouse or Magic Trackpad



iMac 27-inch
with Retina 5K display
from \$1999

- 27-inch (diagonal) LED-backlit Retina display
- 3.3GHz or 3.5GHz quad-core Intel Core i5 or 4.0GHz quad-core Intel Core i7 processor
Turbo Boost up to 4.4GHz
- 1TB or 3TB 7200-rpm hard drive; 1TB or 3TB Fusion Drive; or up to 1TB flash storage (SSD)²
- Apple Wireless Keyboard and Magic Mouse or Magic Trackpad

Mac mini



Mac mini
from \$499

- Up to 3.0GHz dual-core Intel Core i7 processor
Turbo Boost up to 3.5GHz
- Highly energy efficient⁵
- Up to 1TB hard drive; 256GB, 512GB, or 1TB flash storage (SSD); or 1TB or 2TB Fusion Drive²



Mac Pro
from \$2999

- 3.7GHz quad-core, 3.5GHz 6-core, 3.0GHz 8-core, or 2.7GHz 12-core Intel Xeon E5 processor
Up to 30MB of L3 cache
- Up to 1TB flash storage (SSD)²

[Mac](#)[iPad](#)[iPhone](#)[Watch](#)[TV](#)[Music](#)[Support](#)

OS X

[Design](#)[Continuity](#)[Better Apps](#)[What is OS X](#)[How to Upgrade](#)[Upgrade](#)

OS X Yosemite

Every bit as powerful as it looks.

An elegant design that feels entirely fresh, yet inherently familiar. The apps you use every day, enhanced with new features. And a completely new relationship between your Mac and iOS devices. OS X Yosemite changes how you see your Mac.

And what you can do with it. Upgrade for free at the Mac App Store.

OS X El Capitan

A refined experience and improved performance for your Mac.

FREE UPGRADE

[Learn More >](#)

Best New Apps & Games



OS X El Capitan

Utilities

★★★★★ 165 Ratings

GET ▾

OSX El Capitan

A refined experience and improved performance for your Mac.

100 MB of 6.08 GB — 1 hour 58 minutes

PAUSE

[Learn More >](#)

OS X El Capitan for Developers



<https://developer.apple.com/osx/>

Xcode 6



Xcode

The complete toolset for building great apps.



Xcode 7



Xcode

The complete toolset for building great apps.



Swift 2

OS X Yosemite for Developers



iOS 8

for Developers



iOS 8 includes over 4,000 new APIs that let you add amazing new features and capabilities to your apps

iOS 9 for Developers



Developing for iOS 9

iOS 9 SDK includes new APIs and services that enable new categories of apps and features. Multitasking and gaming APIs help enhance app functionality and create immersive games. Expanded search capabilities, and new support for CloudKit, HomeKit, HealthKit, and MapKit extend iOS to more places than ever before.



Xcode 6

The screenshot displays the Xcode 6 development environment. The top menu bar includes Xcode, File, Edit, View, Find, Navigate, Editor, Product, Debug, Source Control, Window, and Help. The status bar shows the project name 'Shutterbugs', the target 'iPhone 5s', and the build status 'Build Shutterbugs: Succeeded | 5/29/14 at 9:33 AM'. The interface is divided into several panes:

- Left Pane (Project Navigator):** Shows the project structure for 'Shutterbugs', including targets, views, controllers, and resources.
- Storyboard:** Displays a prototype cell for a table view. The cell contains a green fractal image, the text '3 comments, 2 likes', and a 'Comment' label. The table view is labeled 'Table View Prototype Content'.
- Right Pane (Code Editor):** Shows the Swift code for 'GamesTableViewController.swift'. The code includes imports for UIKit and CoreShutterbugs, and defines the 'GamesTableViewController' class with methods for game list retrieval and table view data source implementation.
- Bottom Pane (Constraints Panel):** Shows the 'Add New Constraints' dialog box. It displays a width of 320 and a height of 316, with options for equal widths, equal heights, and aspect ratio.

```
// GamesTableViewController.swift
// Shutterbugs
// Copyright (c) 2014 Apple Inc. All rights reserved.

import UIKit
import CoreShutterbugs

class GamesTableViewController: UITableViewController {

    var userSession: UserSession!

    func gameListForSection(section: Int) -> Game[] {

        if section == 0 {
            return self.userSession.inProgressGames as Game[]
        } else {
            return self.userSession.completedGames as Game[]
        }
    }

    // Get the appropriate game (or nil) for the given index path.
    func gameForIndexPath(indexPath: NSIndexPath) -> Game? {

        // Get the appropriate game for this cell.
        var games = self.gameListForSection(indexPath.section)

        if indexPath.row < games.count {
            return games[indexPath.row]
        }

        return nil
    }

    // Table view data source

    override func numberOfSectionsInTableView(tableView:
    UITableView!) -> Int {
        // Return the number of sections.
        return 2
    }

    override func tableView(tableView: UITableView!,
    numberOfRowsInSectionSection section: Int) -> Int {
        // Return the number of rows in the section.
    }
}
```

Get the Tools

Mac App Store



Xcode **FREE** ▼



Download



Xcode 7



Xcode 7

This release includes the Xcode IDE, Swift 2 compiler, Instruments, Simulator, and latest SDKs for OS X, iOS, and watchOS.

Build

7A220

Posted Date

Sep 16, 2015

SDK

iOS 9

OS X v10.11

watchOS 2



Xcode

What's New

Xcode IDE

Interface Builder

Features

Download



Download Xcode for Free



Xcode 7.1 beta 2 PRE-RELEASE

This is the complete Xcode developer toolset for building apps that run on Apple TV, Apple Watch, iPhone, iPad, and Mac. It includes the Xcode IDE, simulators, and all the required tools and frameworks to build apps for iOS, watchOS, tvOS, and OS X.

⬇ [Download Xcode 7.1 beta 2](#)

📄 [Xcode 7.1 beta 2 Release Notes](#)

Build
7B75

Posted Date
Sep 23, 2015

SDK
iOS 9.1 beta
OS X v10.11
watchOS 2
tvOS beta



Xcode 7.0.1

This release includes the Xcode IDE, Swift 2 compiler, Instruments, Simulator, and latest SDKs for OS X, iOS, and watchOS.

[View in the Mac App Store >](#)

Build
7A1001

Posted Date
Sep 28, 2015

SDK
iOS 9
OS X v10.11
watchOS 2



Xcode

Install ▾



Xcode 4+

Essentials

Xcode includes everything developers need to create great applications for Mac, iPhone, iPad, and Apple Watch. Xcode provides developers a unified workflow for user interface design, coding, testing, and debugging. The Xcode IDE combined with the Cocoa frameworks and Swift programming language make developing apps easier and more fun than ever before.

...

...More

What's New in Version 7.0.1

Xcode 7.0.1 fixes bugs related to app thinning.

...

...More

- [Apple Web Site](#) >
- [Xcode Support](#) >
- [App License Agreement](#) >
- [Privacy Policy](#) >

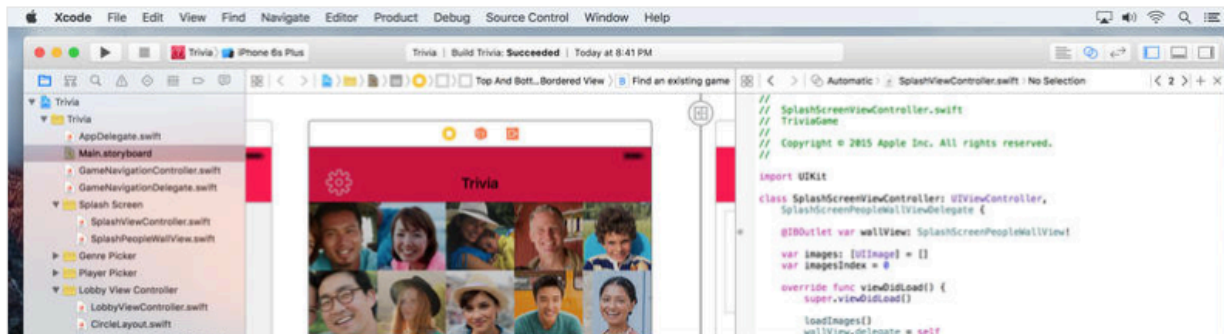
Information

Category: Developer Tools
 Updated: 28 September 2015
 Version: 7.0.1
 Price: Free
 Size: 3.59 GB
 Family Sharing: Yes
 Language: English
 Developer: iTunes S.a.r.l.
 © 1999–2014 Apple Inc.

Rated 4+

Compatibility:
OS X 10.10.4 or later

More Apps by Apple



Xcode





Xcode

Toolbar

Editor area

The screenshot shows the Xcode IDE interface with several components highlighted by red lines and labels:

- Toolbar:** Located at the top left, containing icons for running, debugging, and other development actions.
- Editor area:** The central area displaying Swift code for `AdventureScene.swift`. A breakpoint is set at line 1.1 of the `addSpawnPoints()` function.
- Navigator area:** Located on the left side, showing a project hierarchy for "Adventure" with folders like "Scene", "Assets", and "Particles".
- Debug area:** Located at the bottom center, showing the current state of variables in the debugger, such as `self`, `$yGenerator`, `y`, `location`, and `spot`.
- Utilities area:** Located on the right side, containing panels for "Identity and Type", "Text Settings", "Source Control", and documentation for "View Controller", "Navigation Controller", and "Table View Controller".

Navigator area

Debug area

Utilities area



Xcode

Inspector pane

Identity and Type Show

Interface Builder Document

Opens in ▼

Builds for ▼

View as ▼

Use Auto Layout

Use Size Classes

Inspector bar

Library pane

View Controller - A controller that supports the fundamental view-management model in iOS.

Navigation Controller - A controller that manages navigation through a hierarchy of views.

Table View Controller - A controller that manages a table view.

Library bar

Filter bar

Getting Started with Xcode 7 Development

1. Get a Mac
2. Register as an Apple Developer
3. Install Xcode
4. Enroll in the Apple Developer Program
(Optional)



Swift.

A modern programming language
that is safe, fast, and interactive.



Swift 2

- Swift is a powerful and intuitive **programming language** for **iOS**, **OS X**, and **watchOS**.
- Writing Swift code is interactive and fun, the syntax is concise yet expressive, and apps run lightning-fast.
- Swift is ready for your next project — or addition into your current app — because Swift code works side-by-side with Objective-C.

Swift Language



A new programming language for iOS and OS X.

- Swift is a new **object-oriented programming language** for **iOS and OS X development**.
- Swift is **modern, powerful, expressive,** and **easy to use**.

Swift Language



- **Unified**
A **complete replacement** for both the **C** and **Objective-C** languages.
 - Swift provides full object-oriented features, and includes low-level language primitives such as types, flow control, and operators.
- **Fast**
- **Complete platform**
- **Safe by design**
- **Modern**
- **Interactive**





Swift 2

```
print("Hello world")
```



```
println("Hello world")
```

```
println("Hello World")
```

```
println("Hello World")
```

'println' has been renamed to 'print'

Fix-it Replace "println" with "print"



Swift 2

```
print("Hello world")
```

Swift

```
let count = 10
var price = 23.55

let firstMessage = "Swift is awesome. "
let secondMessage = "What do you think?"
var message = firstMessage + secondMessage

print(message)
```

Objective-C

```
const int count = 10;
double price = 23.55;

NSString *firstMessage = @"Swift is awesome. ";
NSString *secondMessage = @"What do you think?";
NSString *message = [NSString stringWithFormat:@"%s%s",
firstMessage, secondMessage];

NSLog(@"%@", message);
```

```
var s = "Hello" + " World"
```

```
var myVariable = 82  
myVariable = 90  
let myConstant = 82
```

```
let individualScores = [75, 43, 93, 87, 12]
var teamScore = 0
for score in individualScores {
    if score > 60 {
        teamScore += 3
    } else {
        teamScore += 1
    }
}
print(teamScore)
```



```
let individualScores = [75, 43, 93, 87, 12]
var teamScore = 0
for score in individualScores {
    if score > 60 {
        teamScore += 3
    } else {
        teamScore += 1
    }
}
print(teamScore)
```

```
[75, 43, 93, 87, 12]
0

(3 times)

(2 times)

"11\n"
```

Xcode 7

×



Welcome to Xcode

Version 7.0.1 (7A1001)



Get started with a playground

Explore new ideas quickly and easily.



Create a new Xcode project

Start building a new iPhone, iPad or Mac application.



Check out an existing project

Start working on something from an SCM repository.



Show this window when Xcode launches

Xcode 7 Playground

Choose options for your new playground:

Name

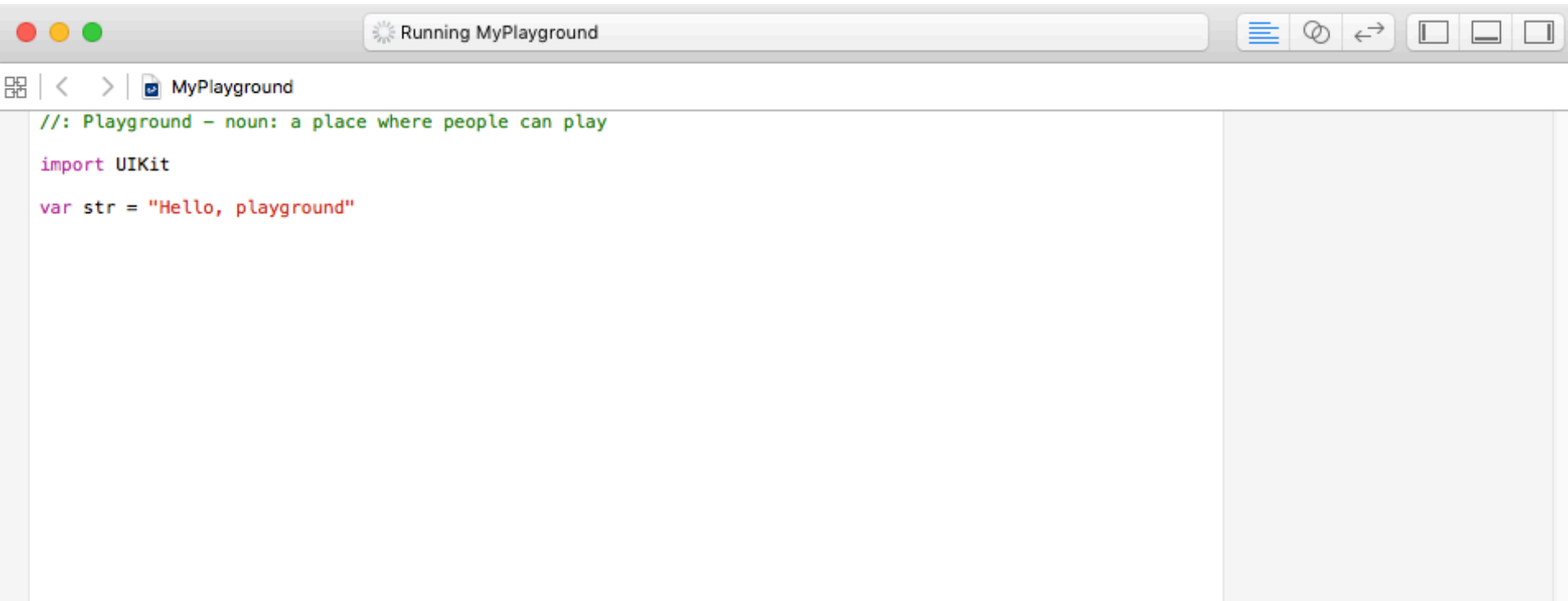
Platform:

Cancel

Previous

Next

Swift 2 in Xcode 7 Playground



Swift 2 in Xcode 7 Playground

A screenshot of the Xcode 7 Playground window. The window title is "MyPlayground" and the status bar shows "Ready | Today at 上午05:04". The code editor contains the following Swift code:

```
//: Playground - noun: a place where people can play
import UIKit
var str = "Hello, playground"
var message = "Hello World"
print(message)
var a = 7
var b = 2
var total = a + b
if (total < 10)
{
    print("Hello Swift 2")
}
```

The right-hand pane shows the output of the code:

```
"Hello, playground"
"Hello World"
"Hello World\n"
7
2
9
"Hello Swift 2\n"
```

The bottom of the window has a play button icon.

Swift 2 in Xcode 7 Playground



Swift 2

```
var message = "Hello World"

print(message)

var a = 7

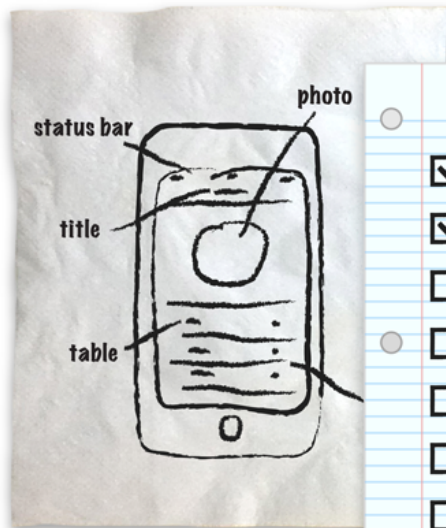
var b = 2

var total = a + b

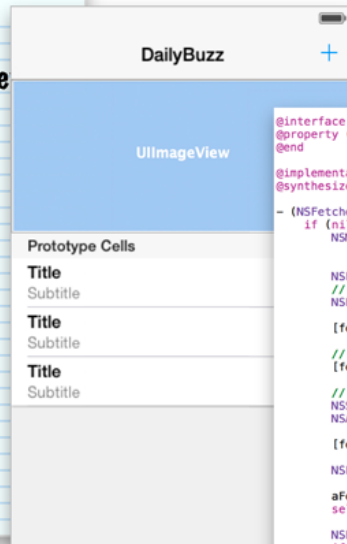
if (total < 10)
{
    print("Hello Swift 2")
}
```

```
"Hello World"
"Hello World\n"
7
2
9
"Hello Swift 2\n"
```

iOS App Development Process



- Focus: Primary Target
- Think top down
- Consistent UI
- Gestures
- Orientation?
- Check target size
- Reduce settings



```
@interface BrewSessionsViewController () <NSFetchedResultsControllerDelegate>
@property (strong, nonatomic) NSFetchedResultsController *fetchedResultsController;
@end

@implementation BrewSessionsViewController
@synthesize fetchedResultsController = _fetchedResultsController;

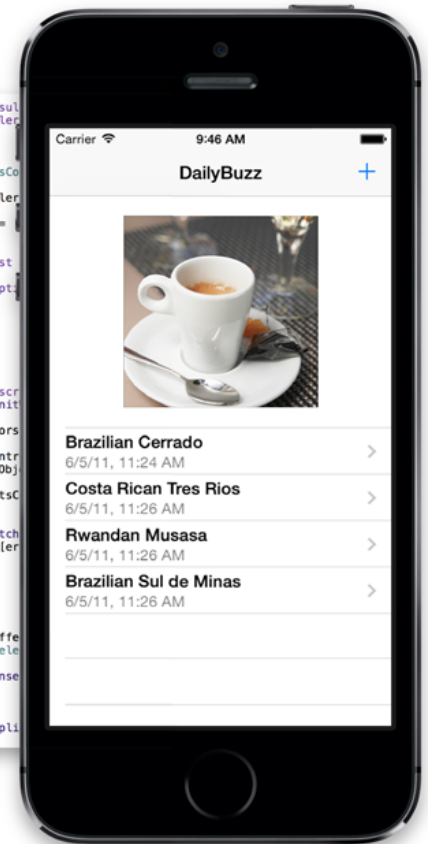
- (NSFetchedResultsController *)fetchedResultsController {
    if (nil == _fetchedResultsController) {
        NSFetchedResultsController *fetchedResultsController =
            [NSFetchedResultsController controllerWithManagedObjectContext:
            managedObjectContext];
    }

    NSFetchedResultsController *aFetchedResultsController =
        fetchedResultsController;
    aFetchedResultsController.delegate = self;
    self.fetchedResultsController = aFetchedResultsController;

    NSError *error = nil;
    if (![self.fetchedResultsController performFetch:
        [NSFetchRequest fetchWithEntity:@"BrewSession"
        managedObjectContext:managedObjectContext
        resultsType:NSFetchRequestResult]] ||
        [error != nil]) {
        NSLog(@"Unresolved error %@, %@", error, [error userInfo]);
        abort();
    }
}

- (void)addBrewSessionForCoffee:(NSManagedObject *)coffee {
    NSFetchedResultsController *aFetchedResultsController =
        fetchedResultsController;
    aFetchedResultsController.delegate = self;
    self.fetchedResultsController = aFetchedResultsController;

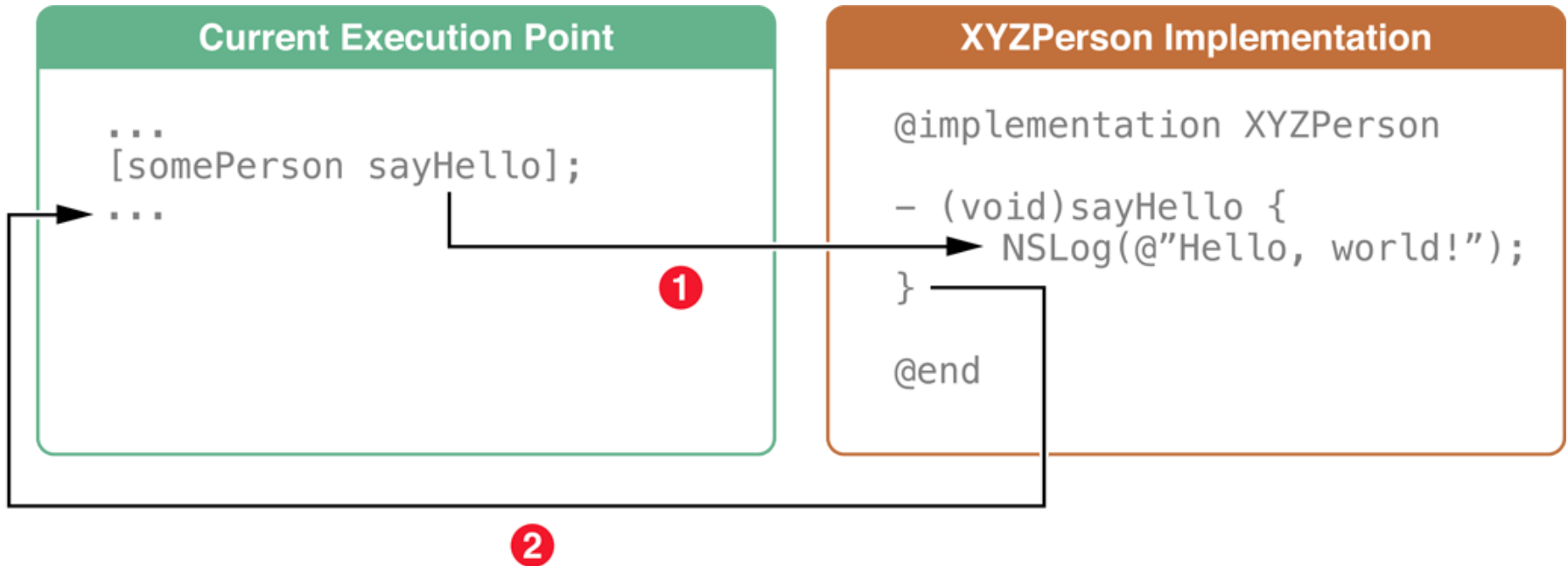
    NSError *error = nil;
    if (![self.fetchedResultsController performFetch:
        [NSFetchRequest fetchWithEntity:@"BrewSession"
        managedObjectContext:managedObjectContext
        resultsType:NSFetchRequestResult]] ||
        [error != nil]) {
        NSLog(@"Unresolved error %@, %@", error, [error userInfo]);
        abort();
    }
}
}
```



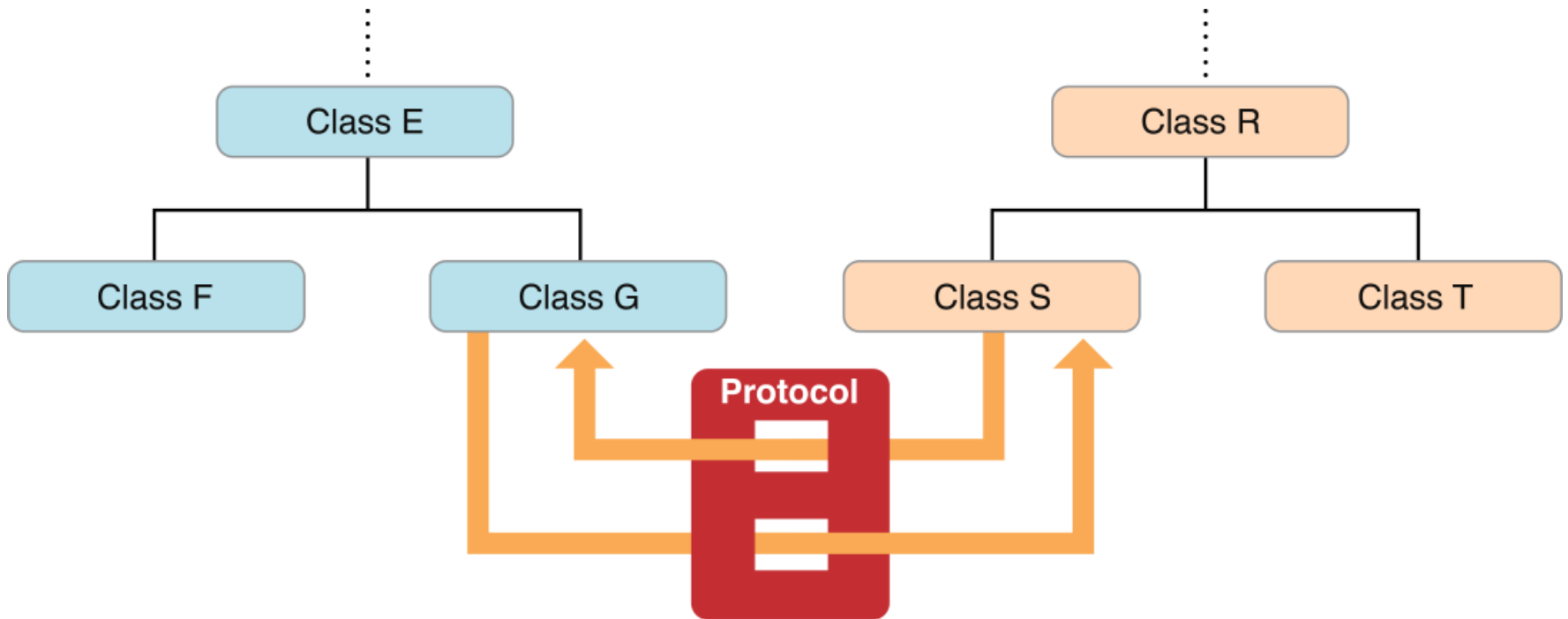
iOS App Development Process

- Defining the Concept
- Designing a User Interface
- Defining the Interaction
- Implementing the Behavior

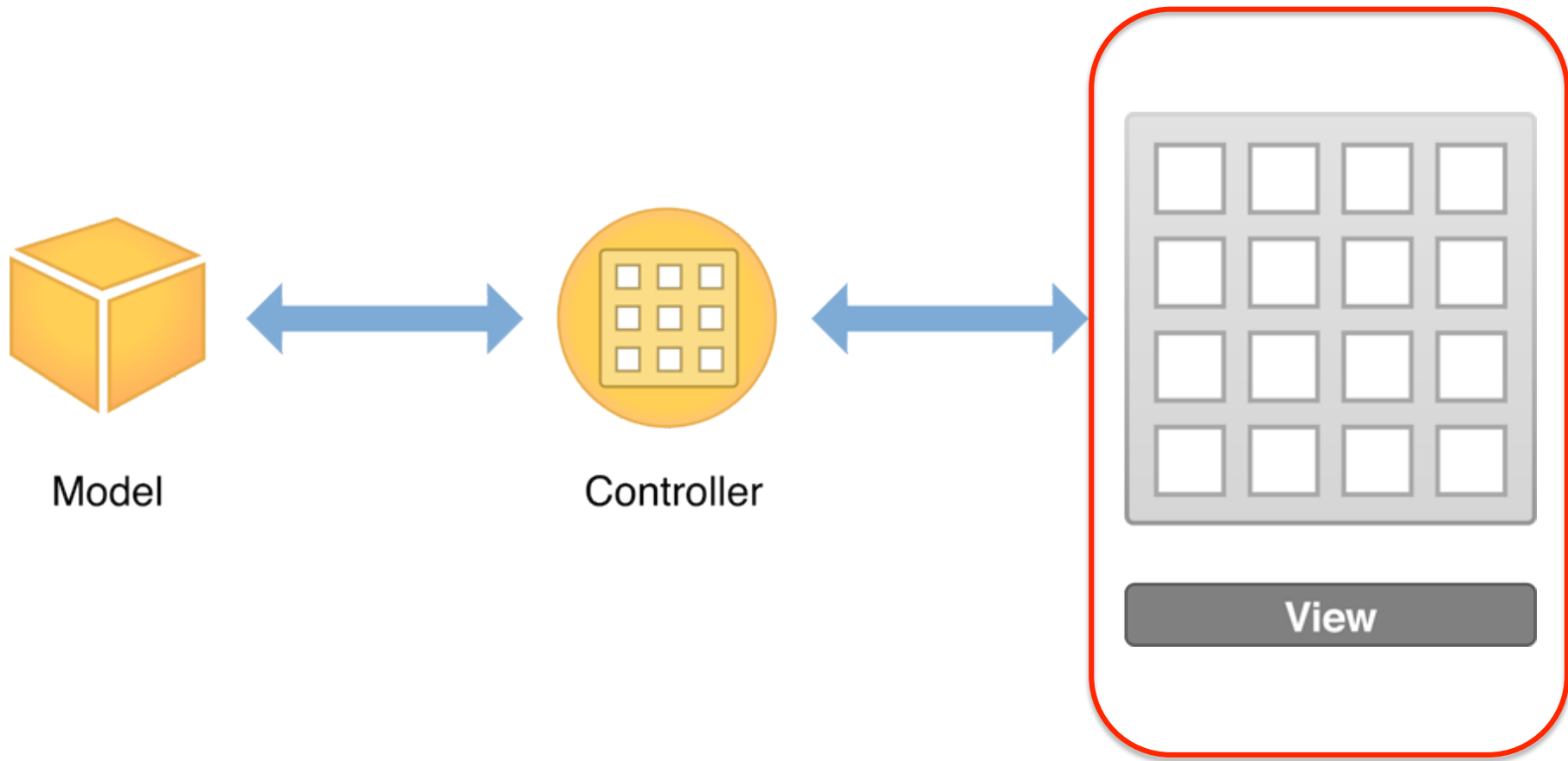
Objects Communicate Through Messages



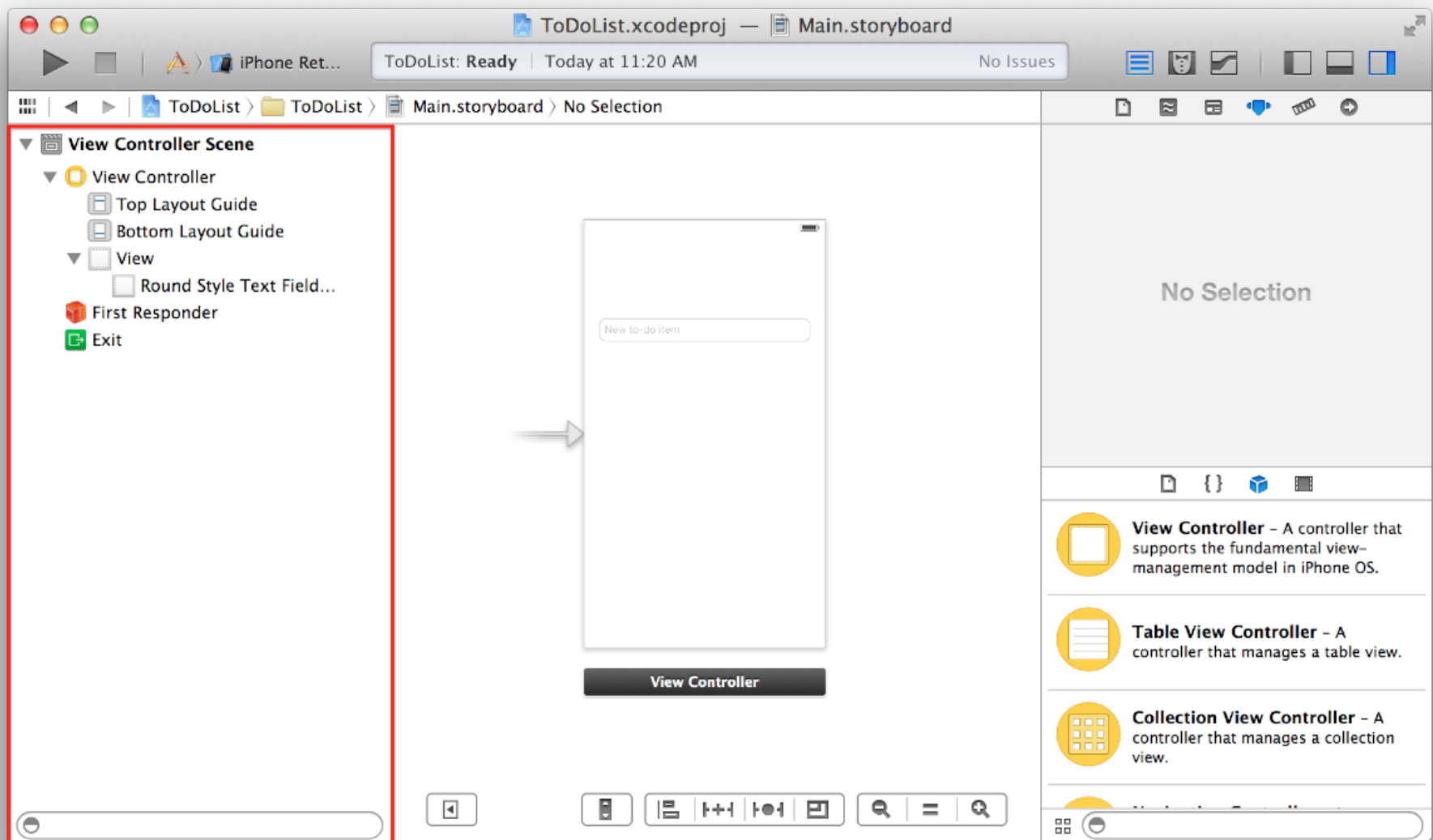
Protocols Define Messaging Contracts



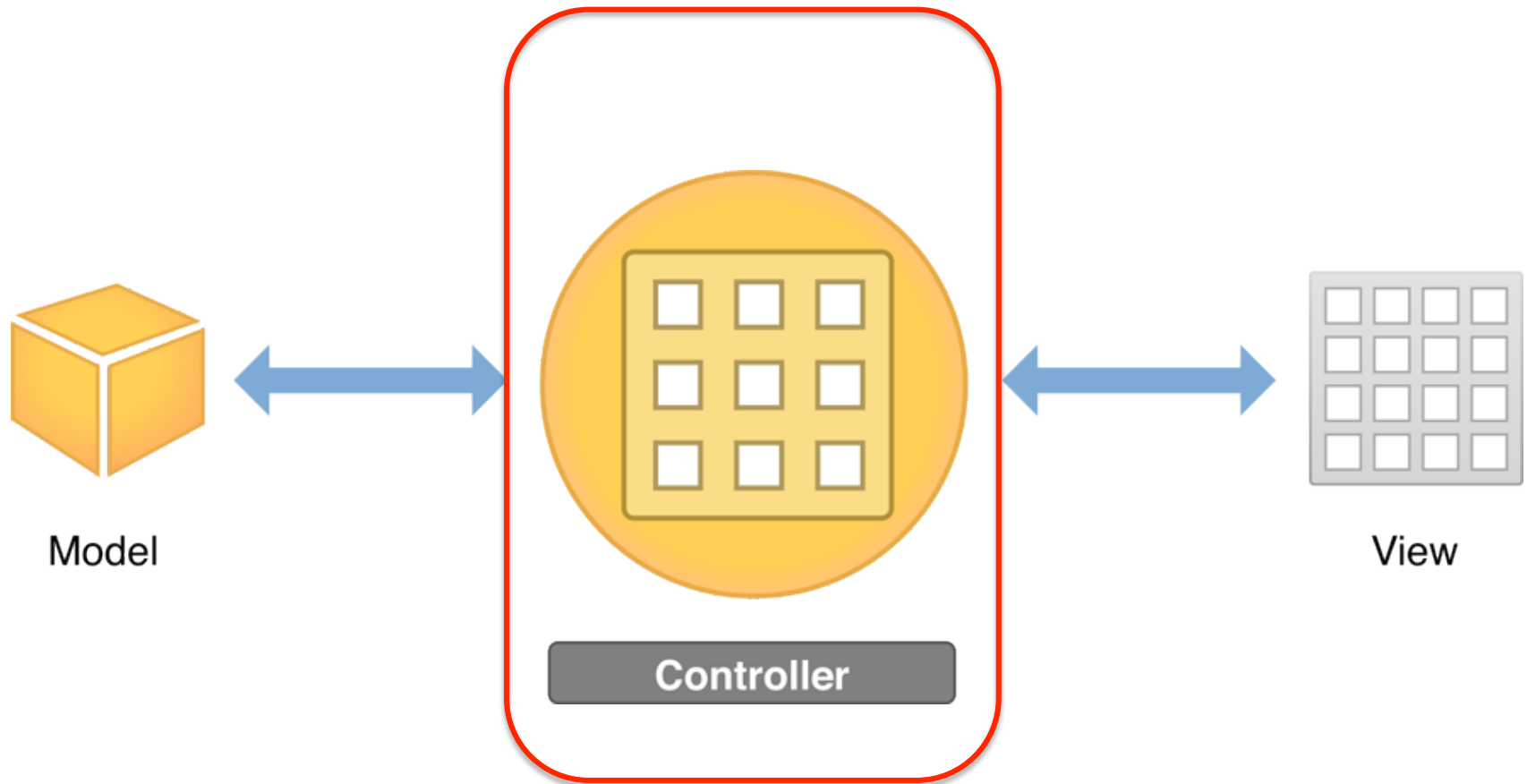
Designing a User Interface



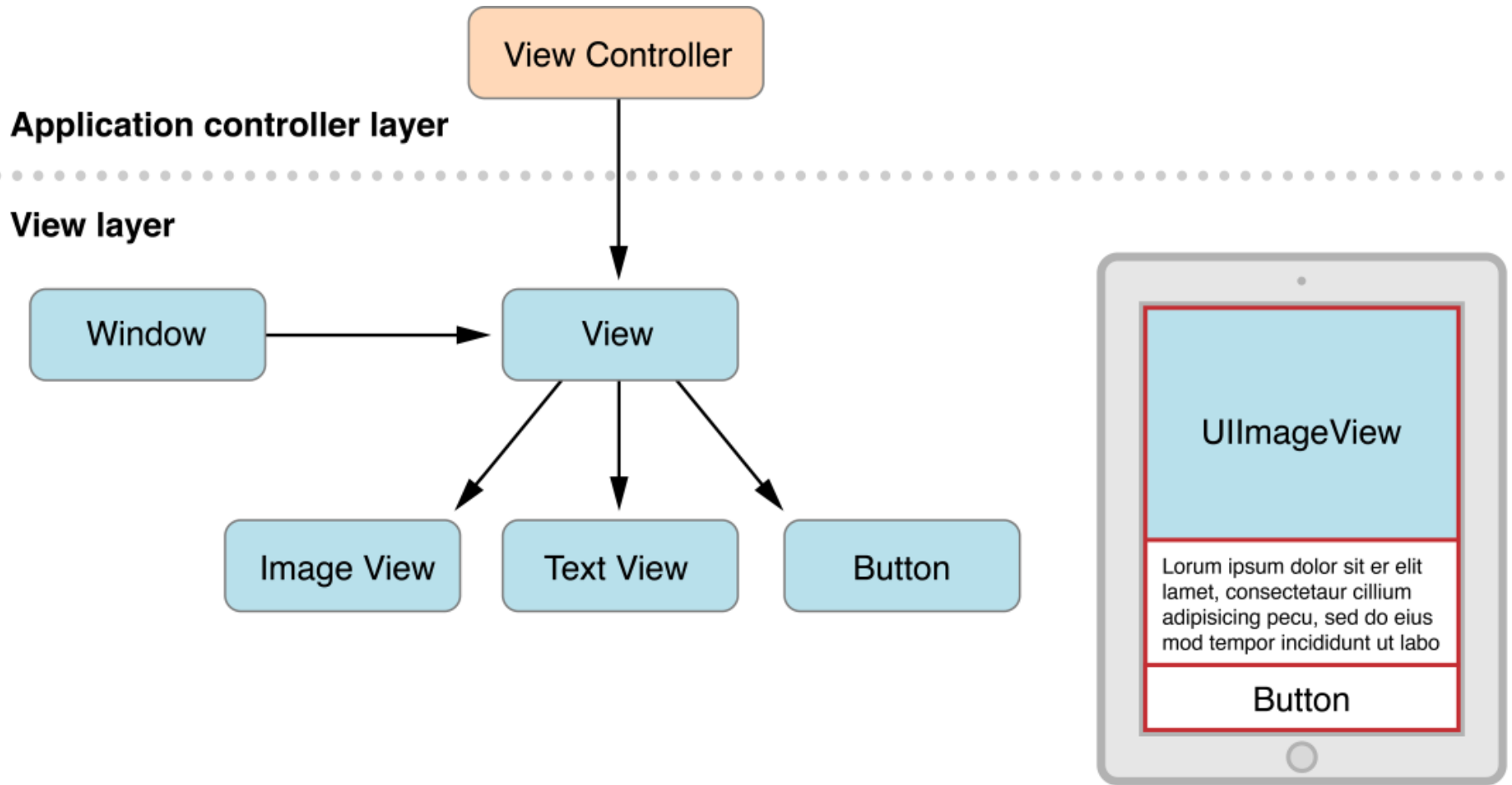
Use Storyboards to Lay Out Views



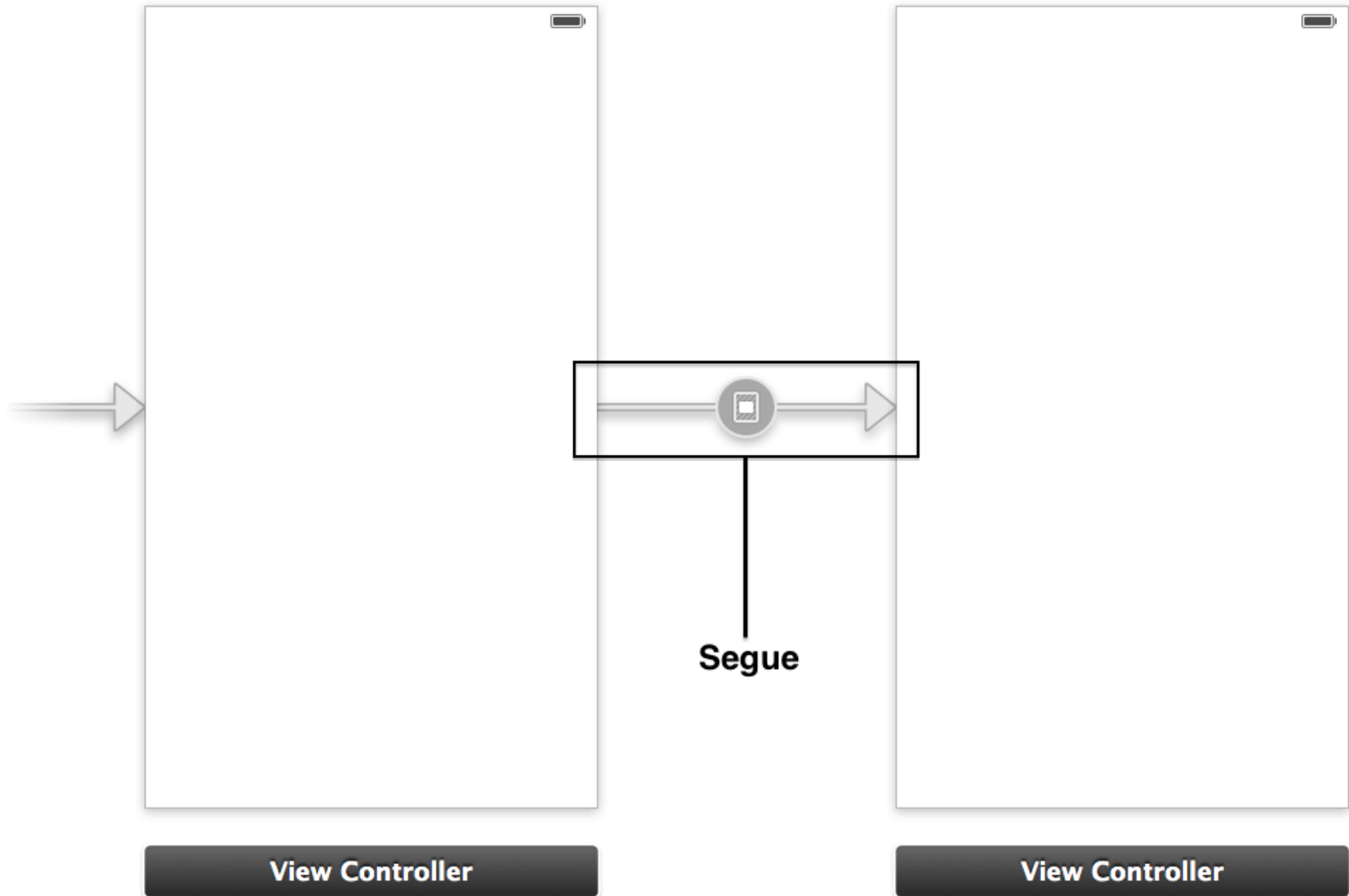
Defining the Interaction



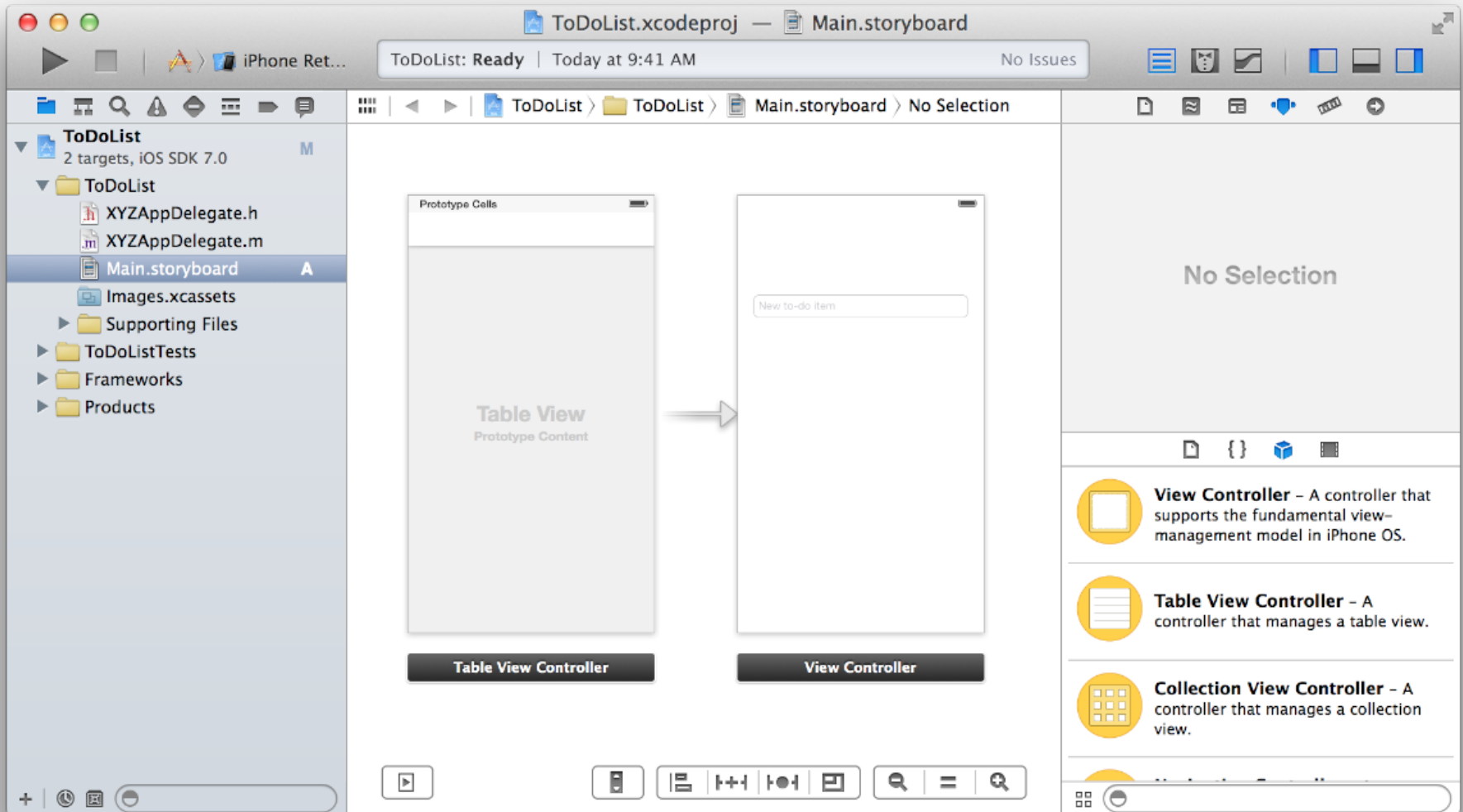
View Controllers



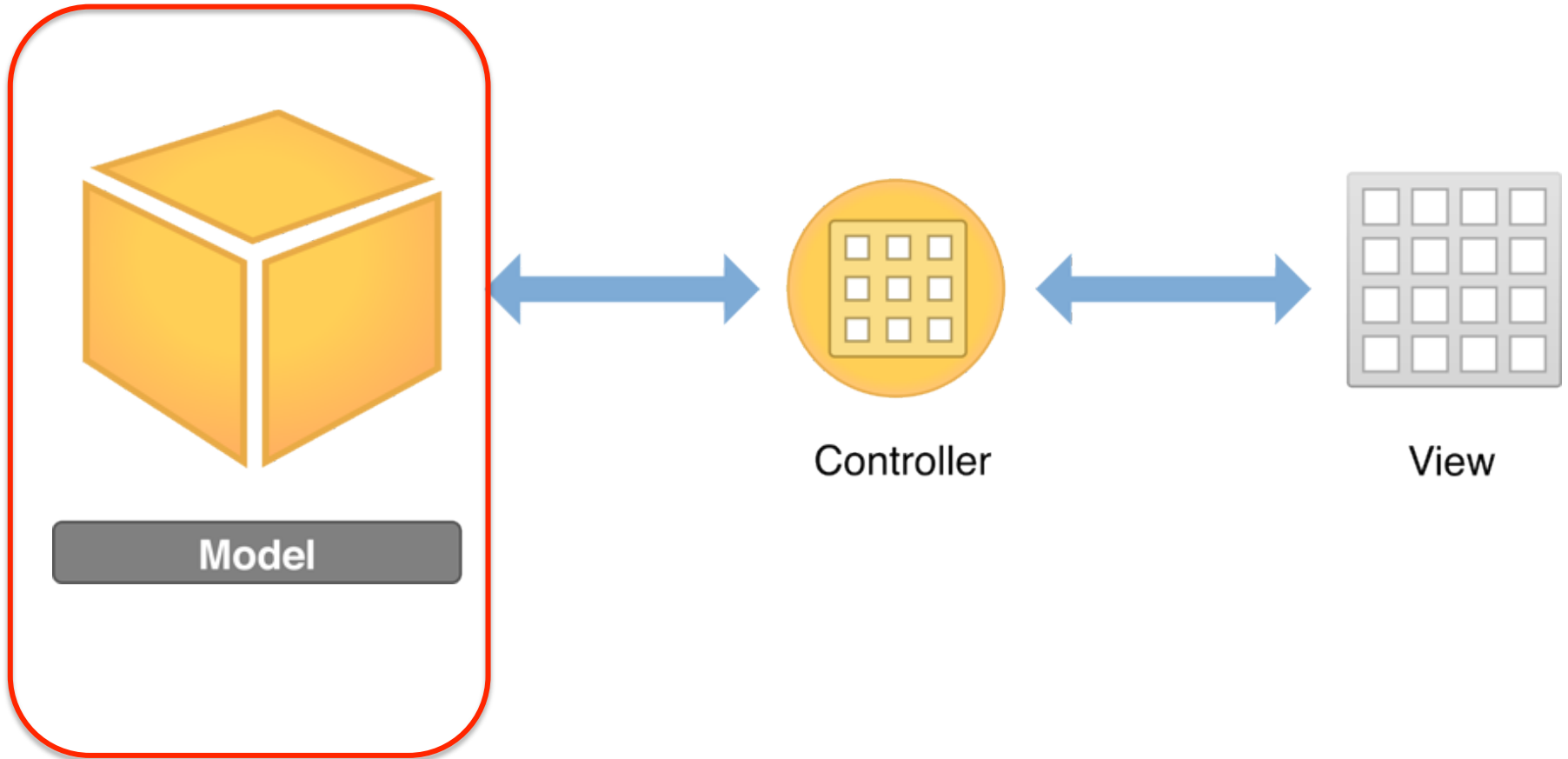
Use Storyboards to Define Navigation



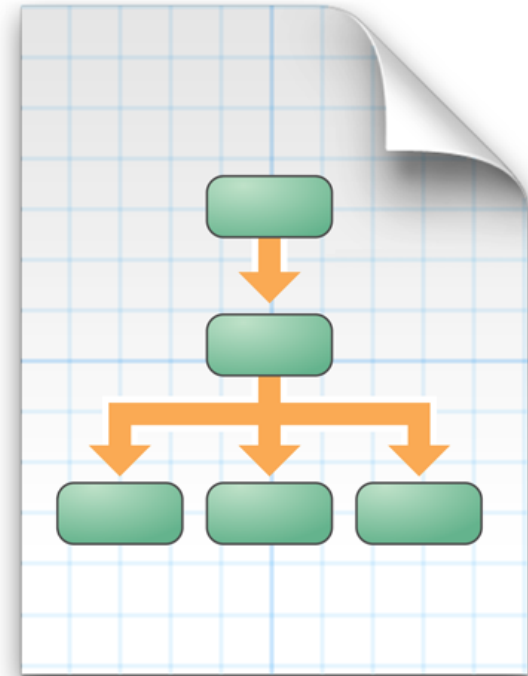
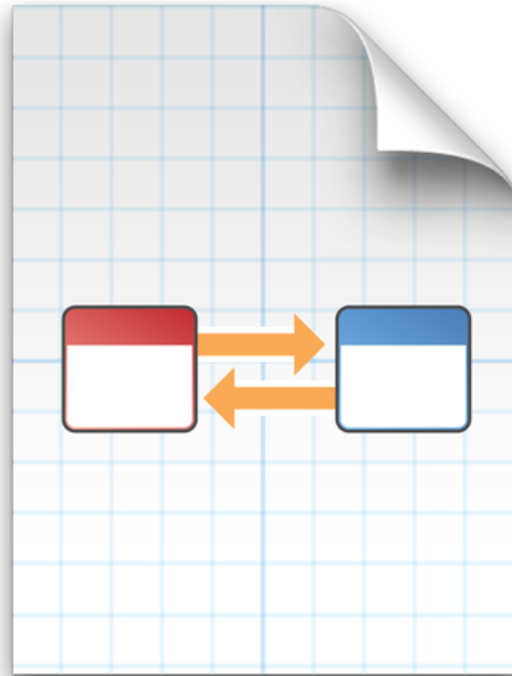
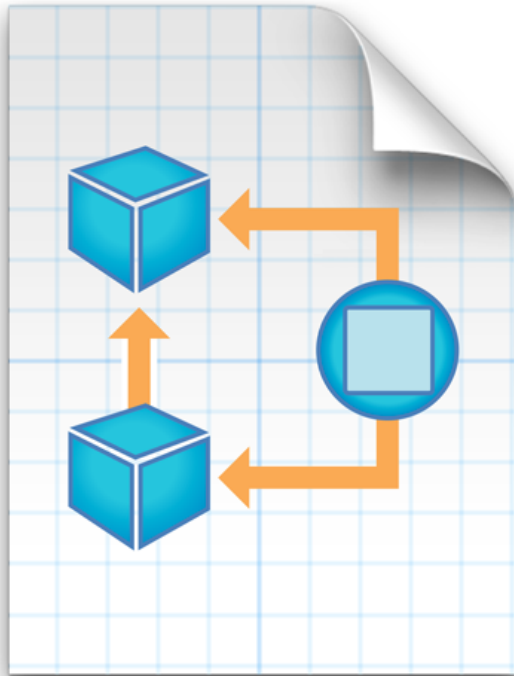
Storyboards



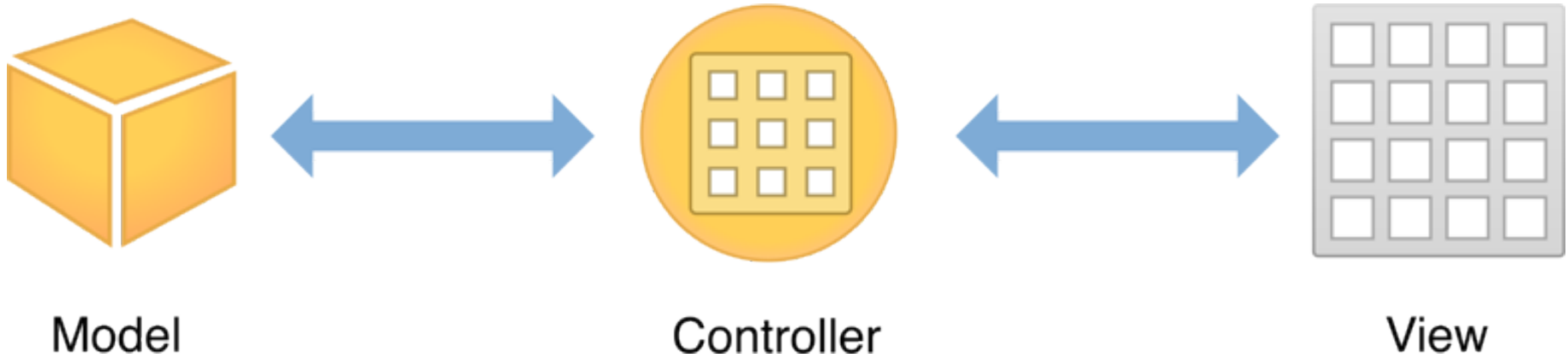
Incorporating the Data



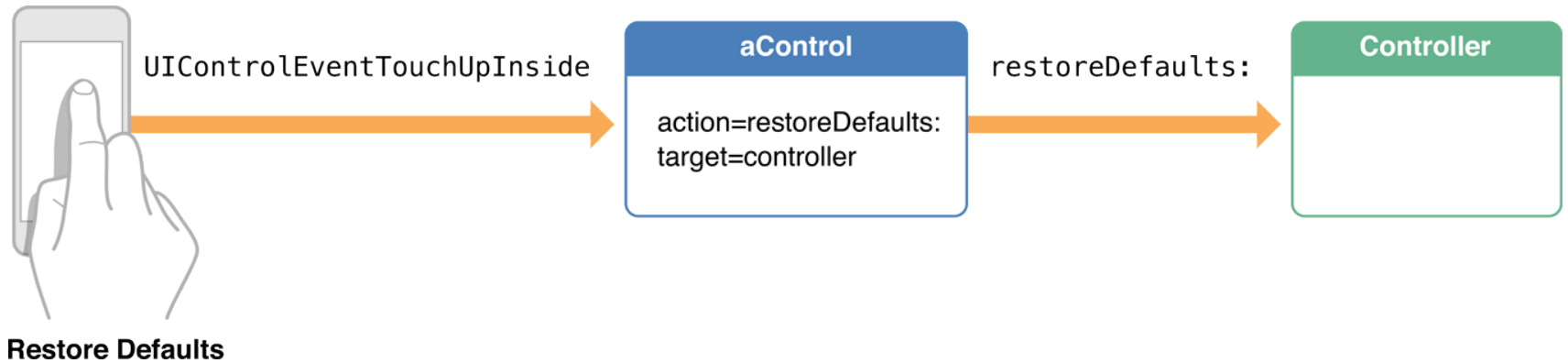
Using Design Patterns



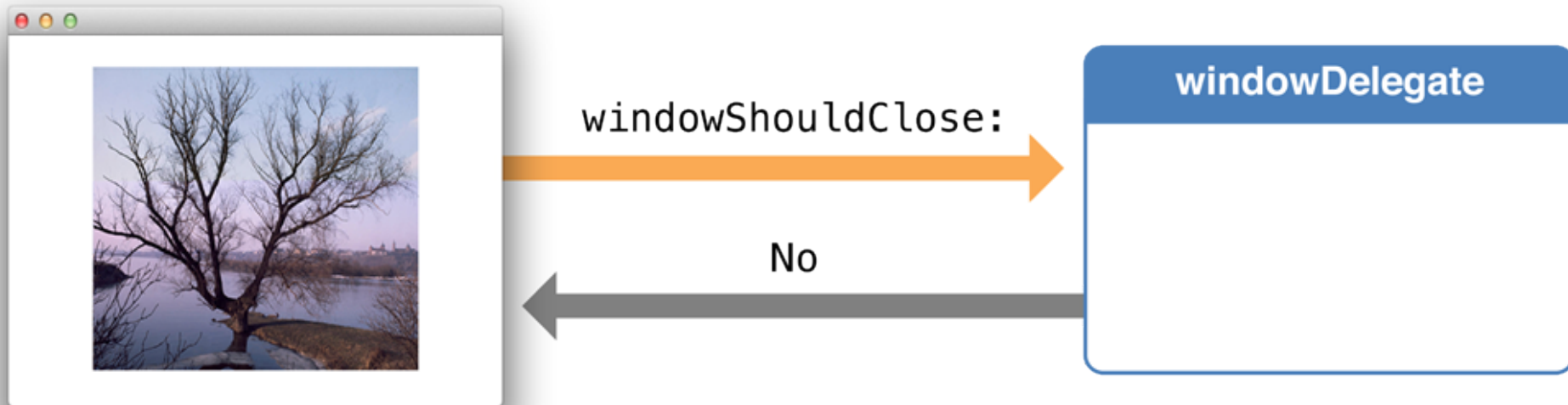
Model-View-Controller (MVC)



Target-Action



Delegation

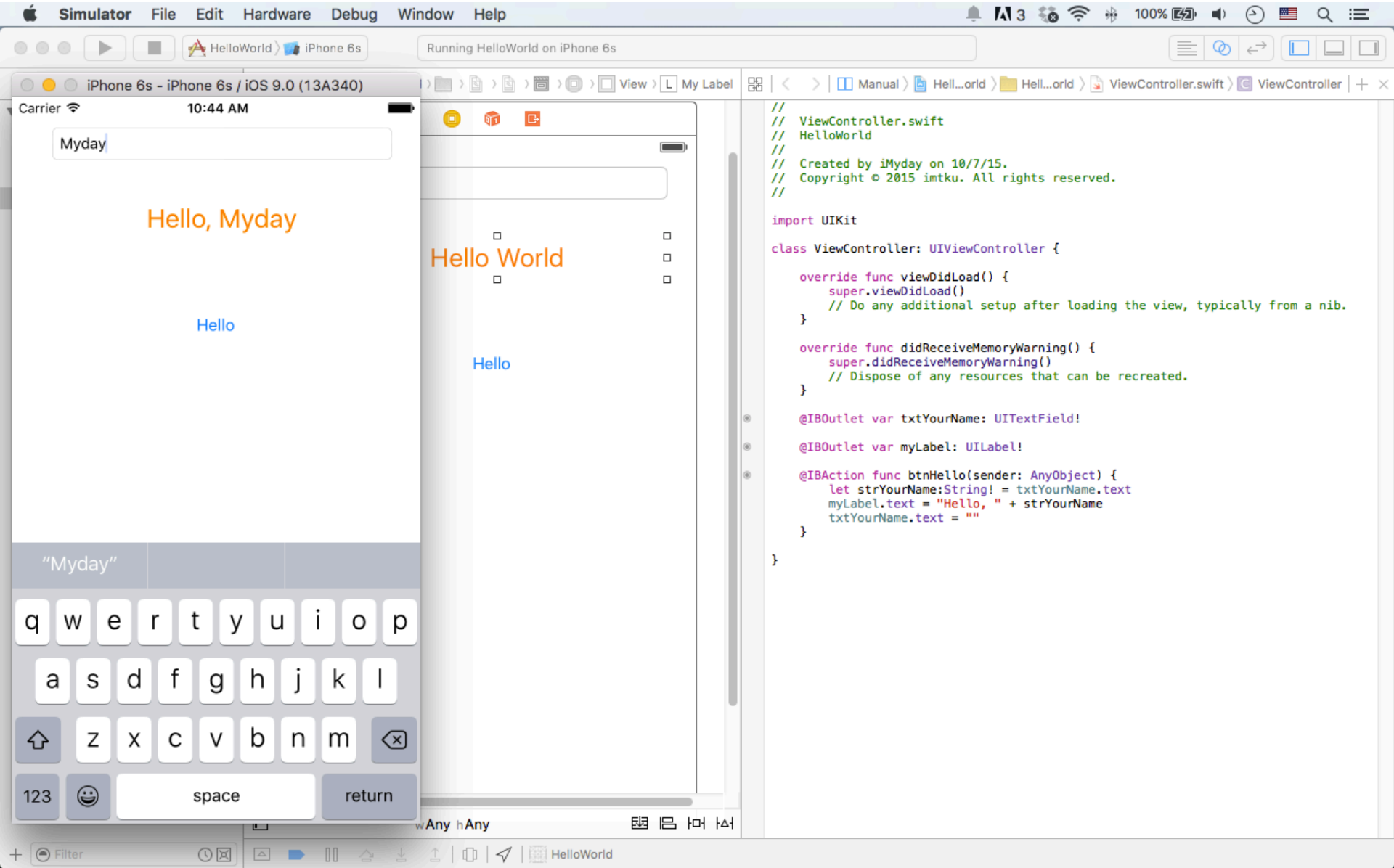


IBOutlet and IBAction

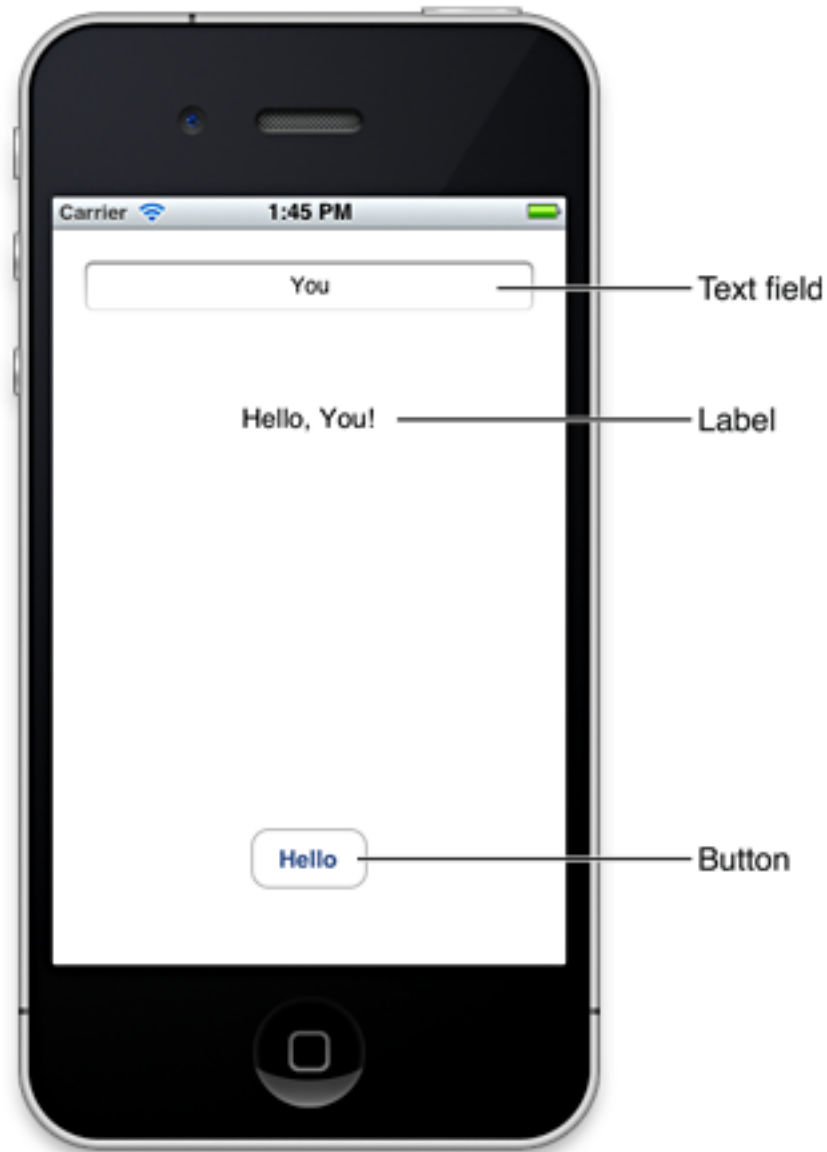
- IBOutlet
 - Interface Builder Outlet
- IBAction
 - Interface Builder Action

Demo:
**Building Your First iOS App with
Xcode 7 (Swift 2)**

Building Your First iOS App with Xcode 7



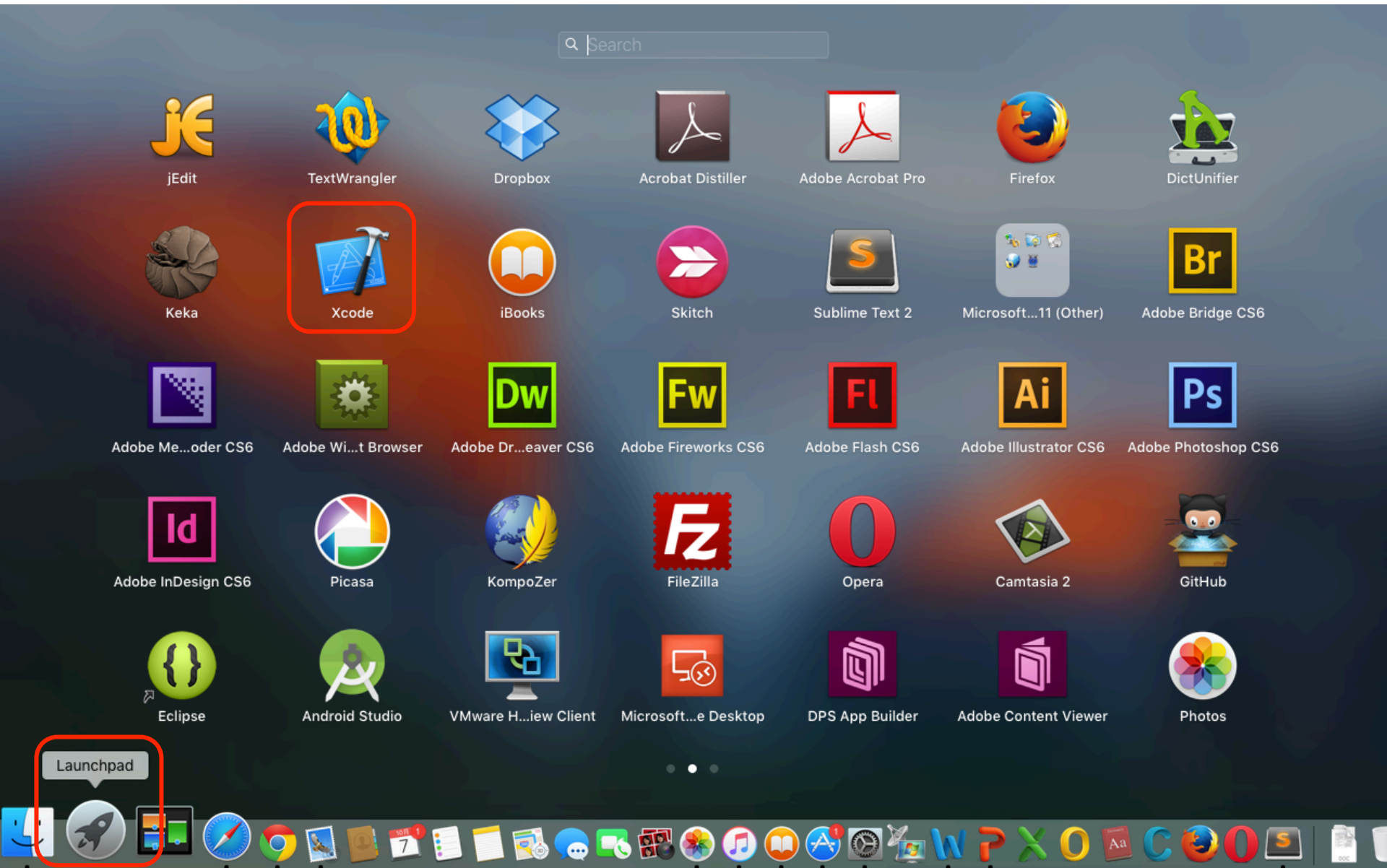
Your First iOS App



Xcode 7 with Swift 2



Launchpad → Xcode



×



Welcome to Xcode

Version 7.0.1 (7A1001)



Get started with a playground

Explore new ideas quickly and easily.



Create a new Xcode project

Start building a new iPhone, iPad or Mac application.



Check out an existing project





Start working on something from an SCM repository.



Show this window when Xcode launches

Project browser sidebar with icons for folders and files.

Choose a template for your new project:

iOS			1	
Application	Master-Detail Application	Page-Based Application	Single View Application	Tabbed Application
Framework & Library				
watchOS				
Application				
Framework & Library	Game			
OS X				
Application				
Framework & Library				
System Plug-in				
Other				

Single View Application

This template provides a starting point for an application that uses a single view. It provides a view controller to manage the view, and a storyboard or nib file that contains the view.

Buttons: Cancel, Previous, Next

Right sidebar area showing "No Selection" and "No Matches" with a search filter input.

Choose options for your new project:

Product Name: HelloWorld

Organization Name: imtku

Organization Identifier: com.imtku

Bundle Identifier: com.imtku.HelloWorld

Language: Swift

Devices: iPhone

- Use Core Data
- Include Unit Tests
- Include UI Tests

Cancel

Previous

Next

No Selection

No Matches

Filter

Swift Language

Choose options for your new project:

Product Name: HelloWorld

Organization Name: imtku

Organization Identifier: com.imtku

Bundle Identifier: com.imtku.HelloWorld

Language: Swift
 Objective-C

Devices: iPhone

- Use Core Data
- Include Unit Tests
- Include UI Tests

Cancel

Previous

Next



iOSApp

Search

Name	Date Modified

Source Control: Create Git repository on My Mac

Xcode will place your project under version control

New Folder Options Cancel Create

No Selection

No Matches

Filter

File Explorer

- HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
- HelloWorldTests
- HelloWorldUITests
- Products

General Capabilities Resource Tags Info Build Settings Build Phases Build Rules

PROJECT

- HelloWorld

TARGETS

- HelloWorld
- HelloWorldTests
- HelloWorldUITests

Identity

Bundle Identifier: com.imtku.HelloWorld

Version: 1.0

Build: 1

Team: None

Deployment Info

Deployment Target: 9.0

Devices: iPhone

Main Interface: Main

Device Orientation:

- Portrait
- Upside Down
- Landscape Left
- Landscape Right

Status Bar Style: Default

Hide status bar

Requires full screen

App Icons and Launch Images

App Icons Source: AppIcon

Launch Images Source: Use Asset Catalog

Launch Screen File: LaunchScreen

Quick Help

No Quick Help

No Matches

Filter

File Explorer (Left Panel):

- ▼ HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
- ▼ HelloWorldTests
- ▼ HelloWorldUITests
- ▼ Products

Target Configuration (Right Panel):

PROJECT

- HelloWorld

TARGETS

- HelloWorld (Selected)
- HelloWorldTests
- HelloWorldUITests

Deployment Info

- Deployment Target: 9.0
- Device: iPhone (Selected dropdown menu)
 - iPhone
 - iPad
 - Universal
- Main Interface: Universal
- Device Orientation:
 - Portrait
 - Upside Down
 - Landscape Left
 - Landscape Right
- Status Bar Style: Default
 - Hide status bar
 - Requires full screen

App Icons and Launch Images

- App Icons Source: AppIcon
- Launch Images Source: Use Asset Catalog
- Launch Screen File: LaunchScreen

Embedded Binaries

Add embedded binaries here

+ -

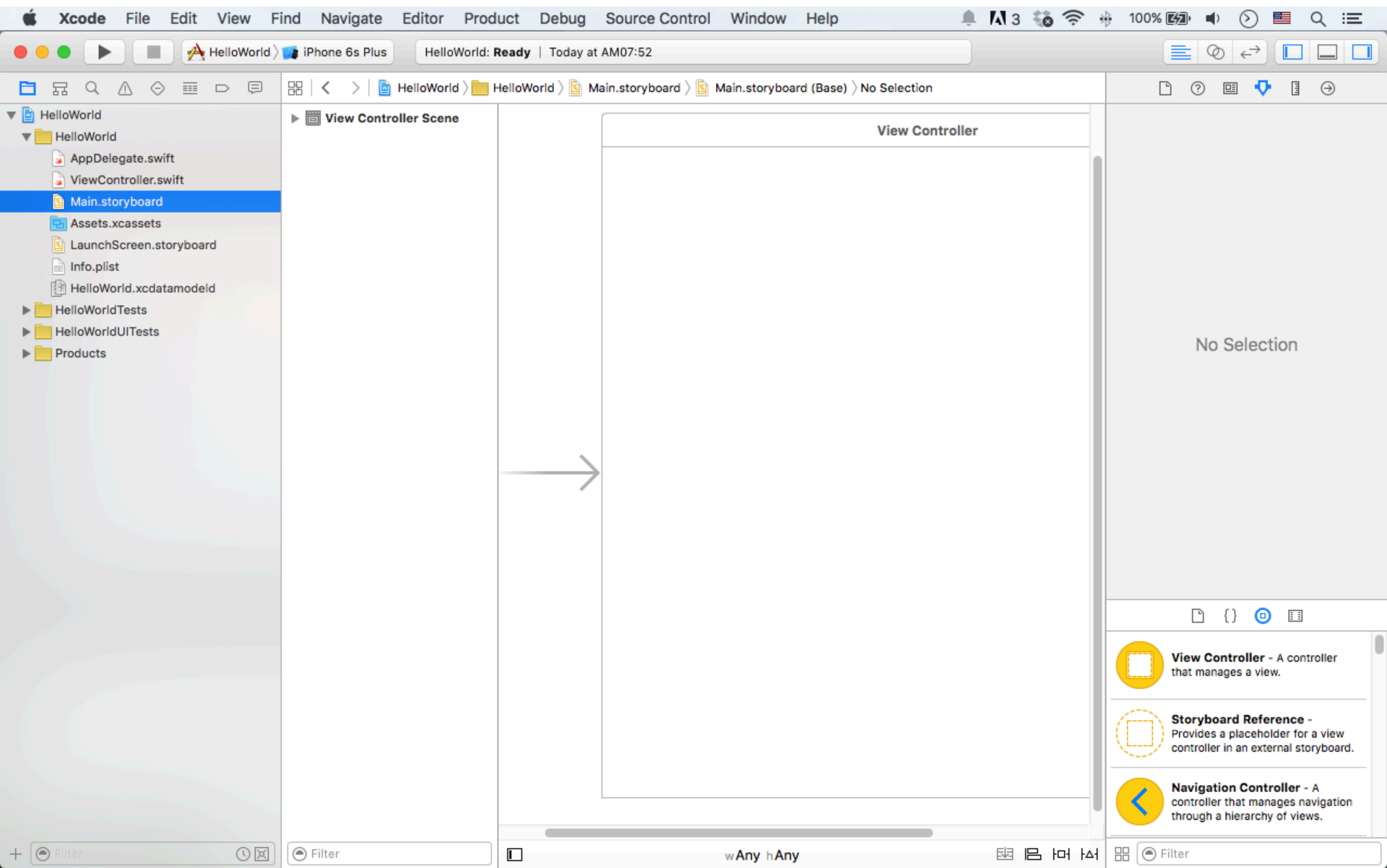
Linked Frameworks and Libraries

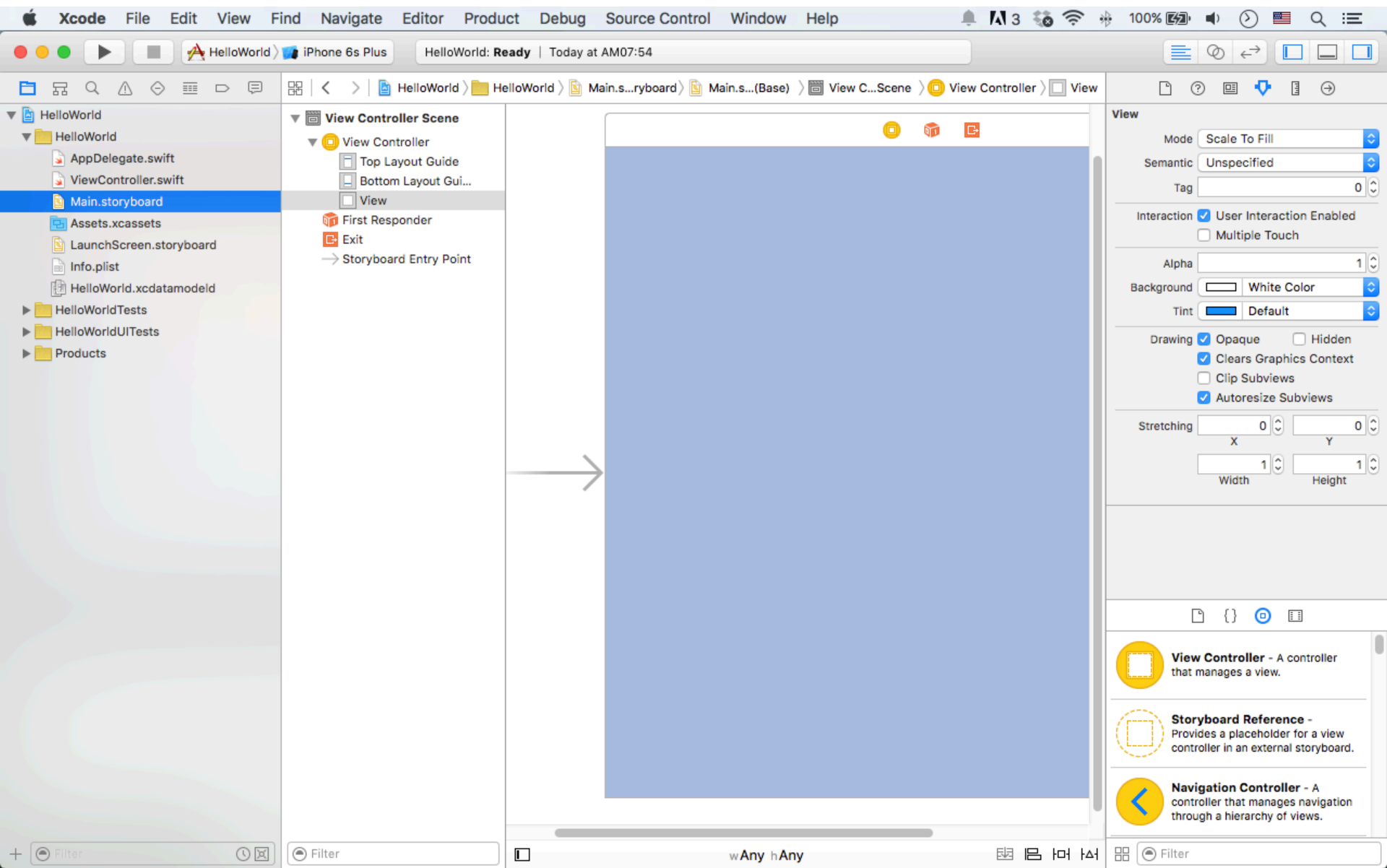
Name	Status
Add frameworks & libraries here	

Quick Help (Far Right Panel):

No Quick Help

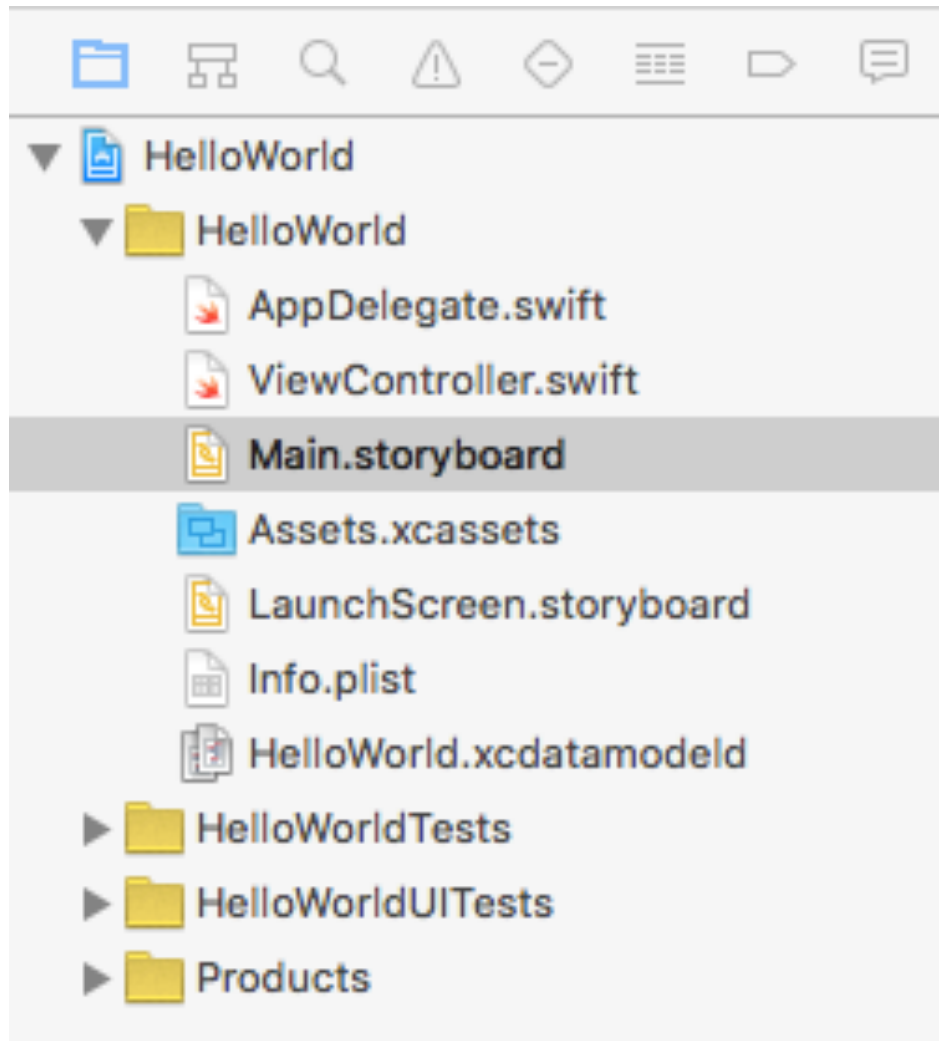
No Matches



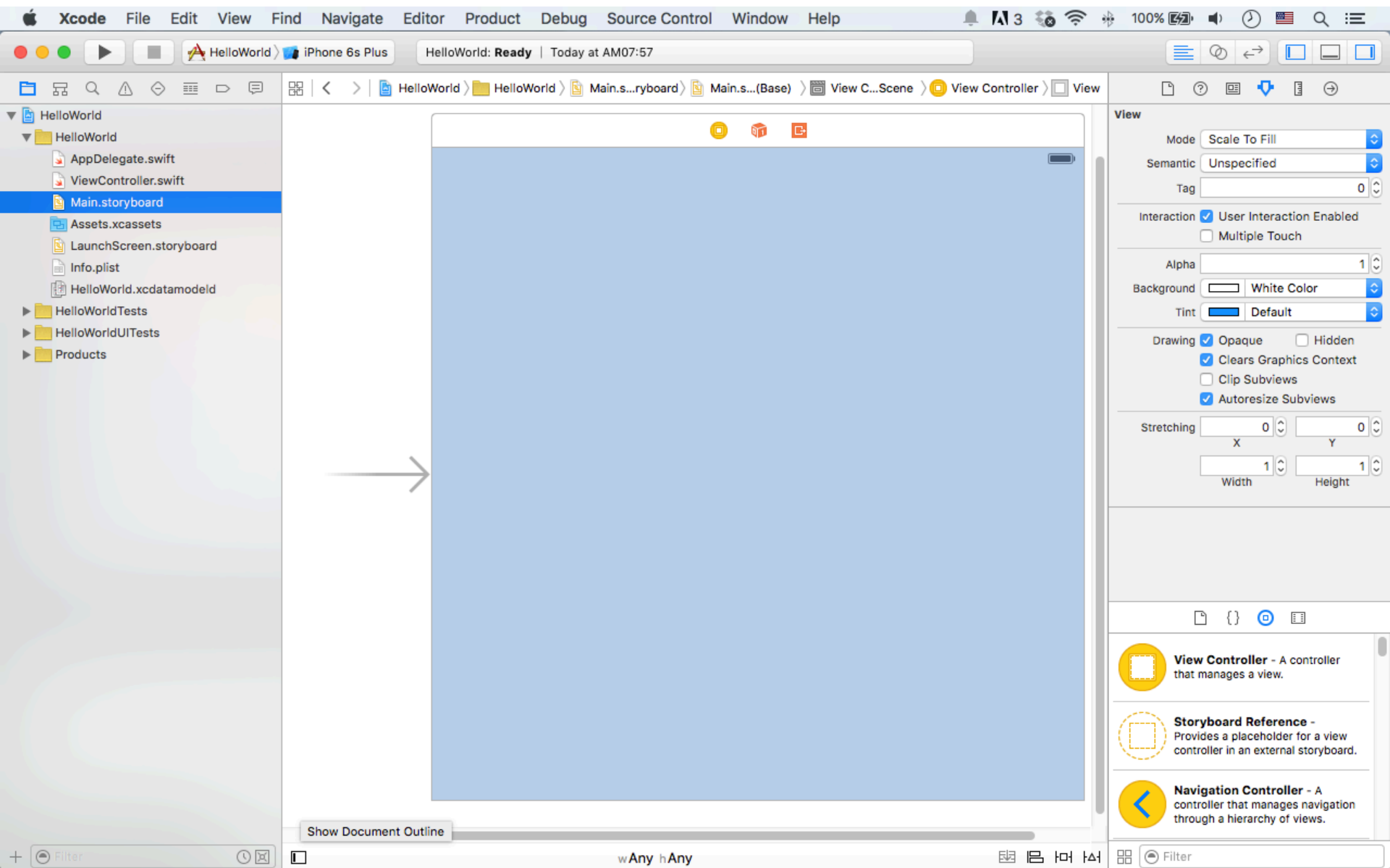


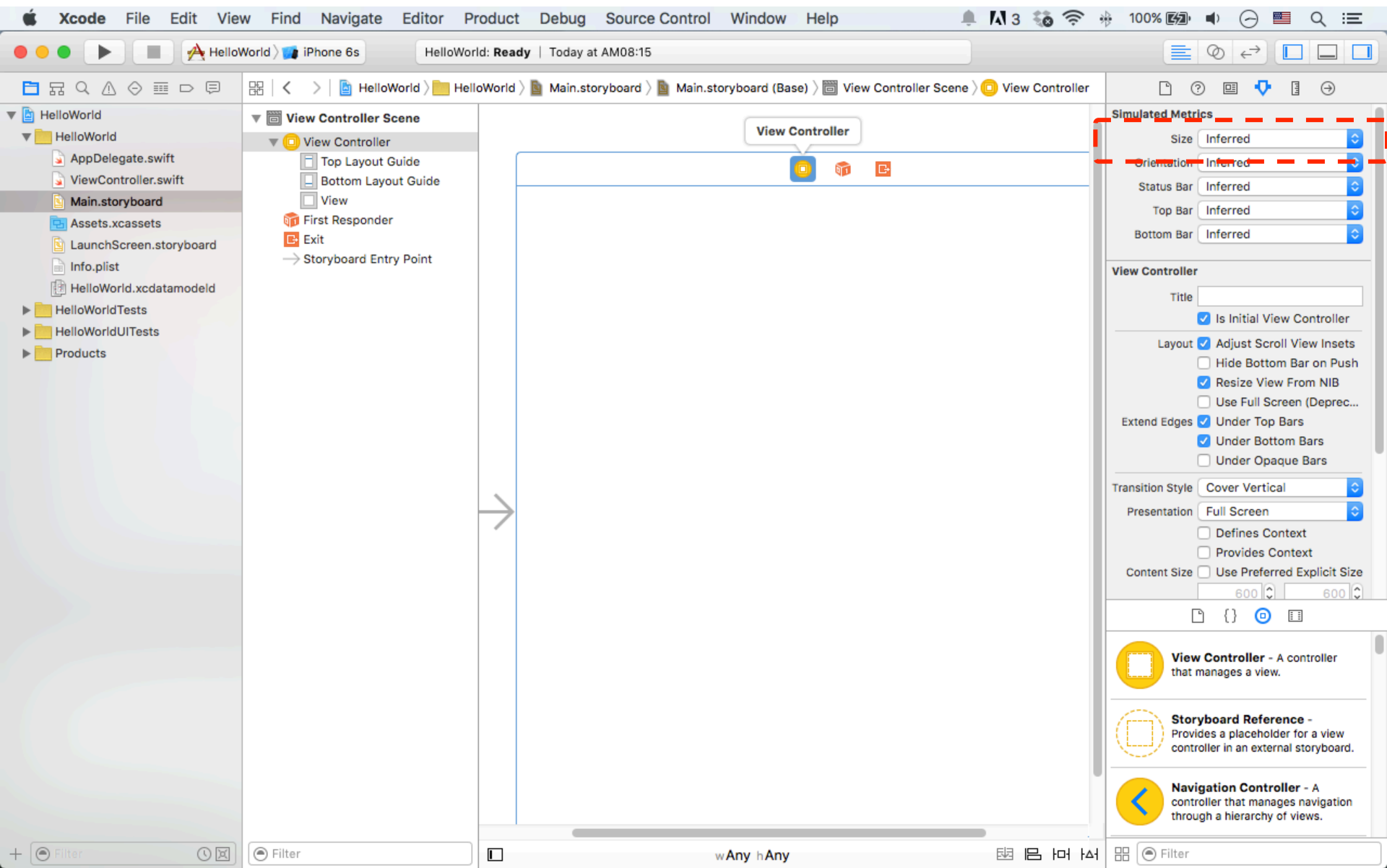
Main.storyboard (UI)

ViewController.swift (Code)



Main.storyboard (UI)



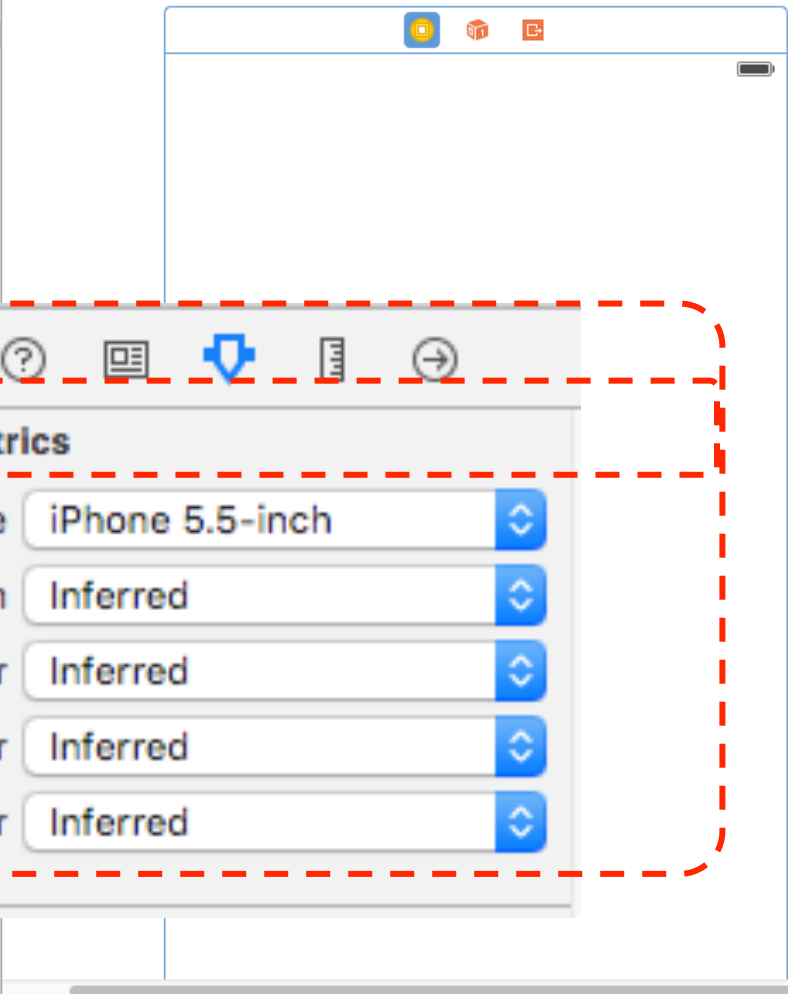


Project Navigator showing file structure for HelloWorld:

- AppDelegate.swift
- ViewController.swift
- Main.storyboard
- Assets.xcassets
- LaunchScreen.storyboard
- Info.plist
- HelloWorld.xcdatamodeld
- HelloWorldTests
- HelloWorldUITests
- Products

View Controller Scene hierarchy:

- View Controller
 - Top Layout Guide
 - Bottom Layout Guide
 - View
 - First Responder
 - Exit
 - Storyboard Entry Point



Simulated Metrics panel:

- Size: iPhone 5.5-inch
- Orientation: Inferred
- Status Bar: Inferred
- Top Bar: Inferred
- Bottom Bar: Inferred

Inspector panel showing Simulated Metrics:

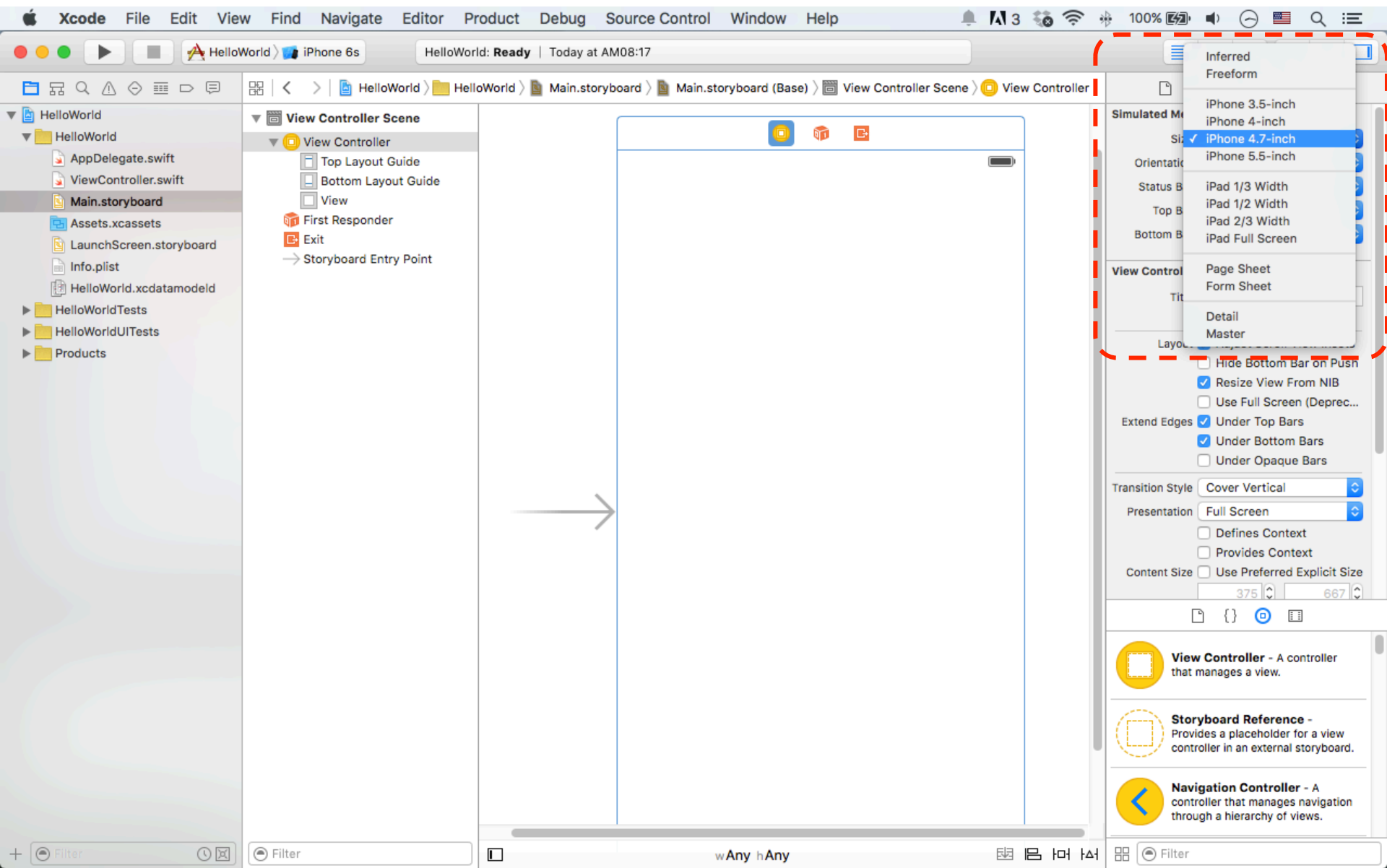
- Simulated Metrics
 - Size: iPhone 5.5-inch (selected)
 - Orientation: Inferred
 - Status Bar: Inferred
 - Top Bar: Inferred
 - Bottom Bar: Inferred

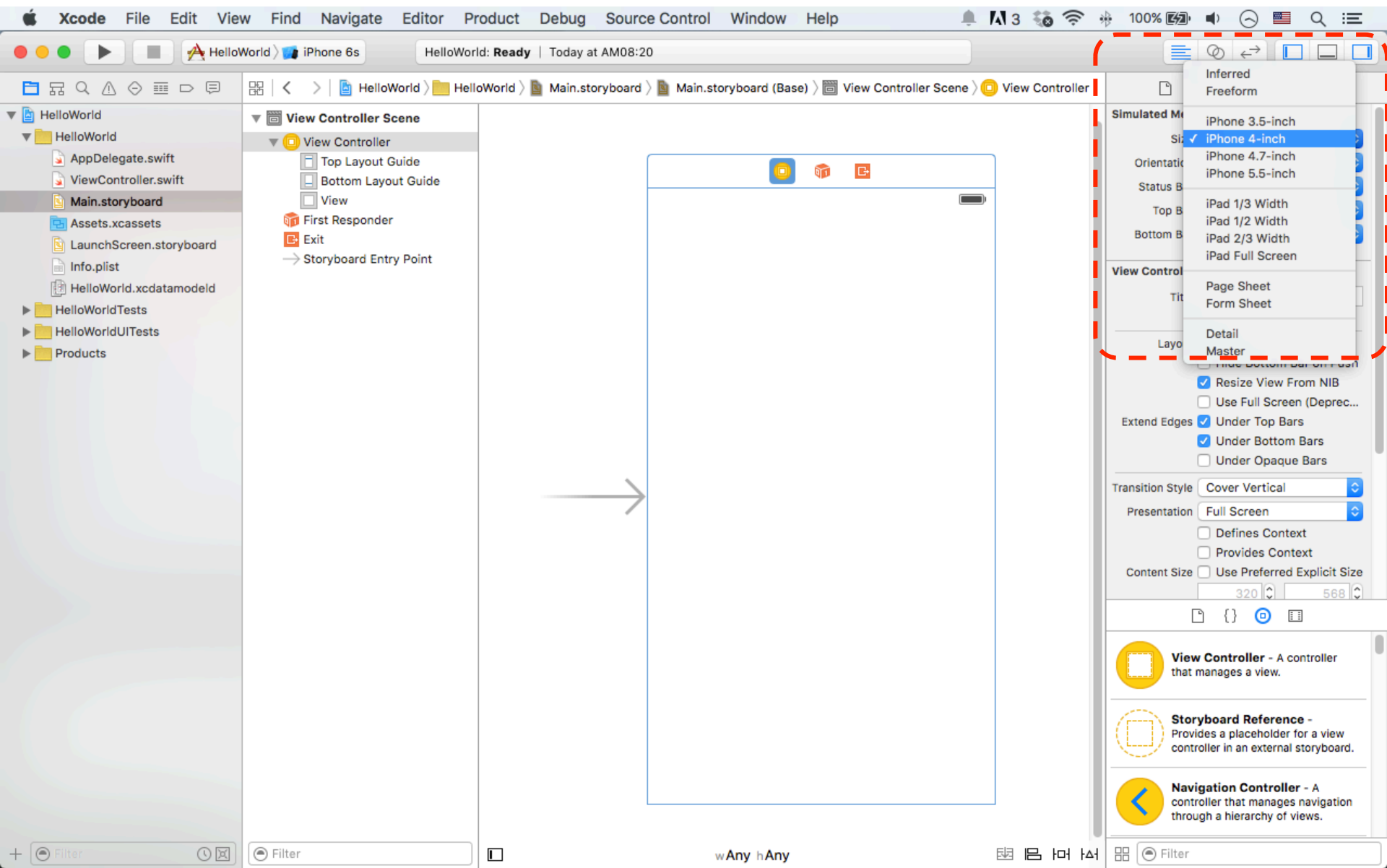
Inspector panel showing Layout and Transition Style settings:

- Layout
 - Adjust Scroll View Insets: checked
 - Hide Bottom Bar on Push: unchecked
 - Resize View From NIB: checked
 - Use Full Screen (Deprecated): unchecked
- Extend Edges
 - Under Top Bars: checked
 - Under Bottom Bars: checked
 - Under Opaque Bars: unchecked
- Transition Style: Cover Vertical
- Presentation: Full Screen
- Content Size: 414 x 736

Inspector panel showing View Controller and Storyboard Reference descriptions:

- View Controller** - A controller that manages a view.
- Storyboard Reference** - Provides a placeholder for a view controller in an external storyboard.
- Navigation Controller** - A controller that manages navigation through a hierarchy of views.





File Explorer (Left):

- ▼ HelloWorld
 - ▼ HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
 - ▼ HelloWorldTests
 - ▼ HelloWorldUITests
 - ▼ Products

Storyboard Outline (Middle-Left):

- View Controller Scene
 - View Controller
 - Top Layout Guide
 - Bottom Layout Guide
 - View
 - First Responder
 - Exit
 - Storyboard Entry Point

Storyboard Canvas (Center):

Properties Inspector (Right):

Simulated Metrics

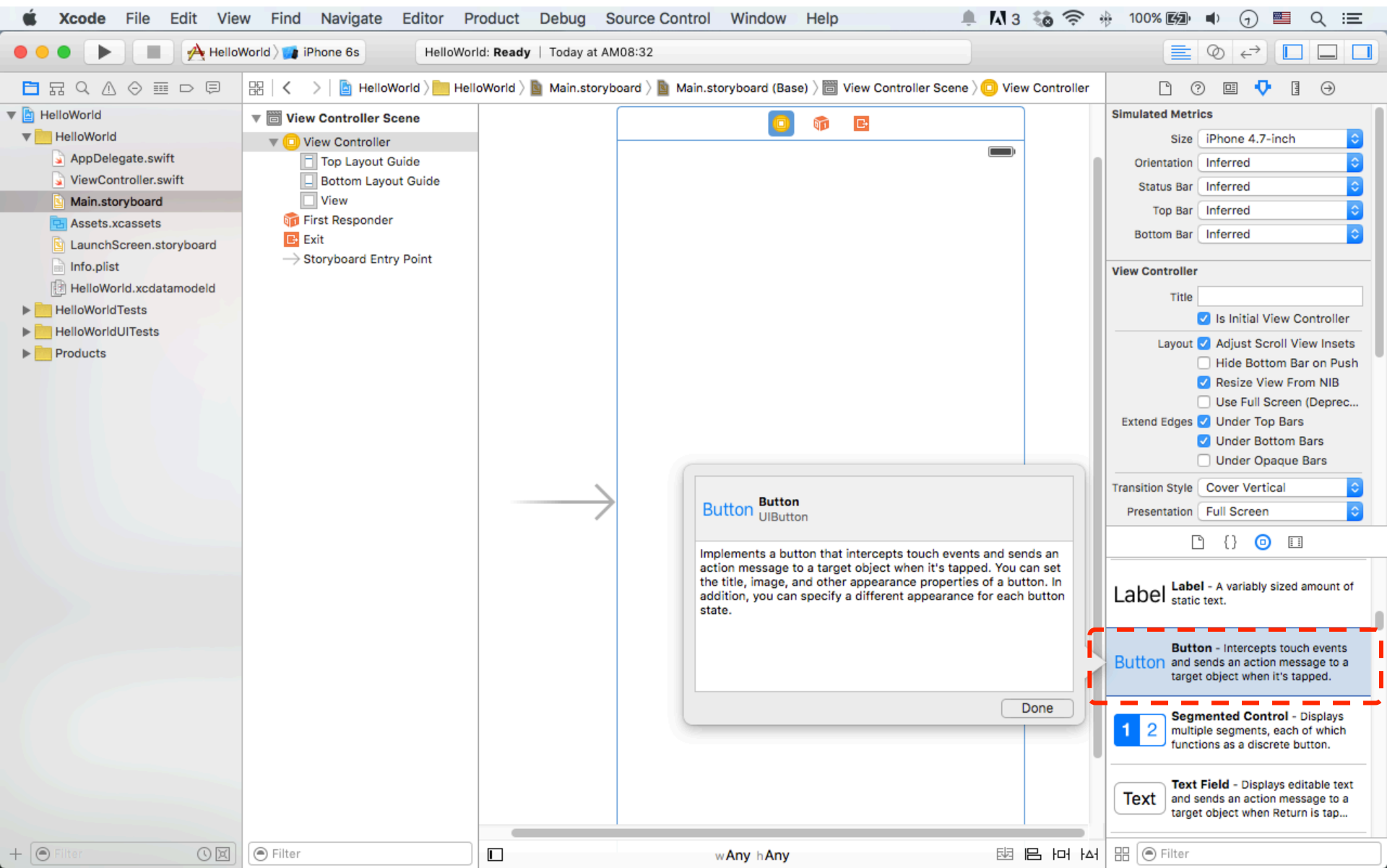
- Size: iPhone 4.7-inch
- Orientation: Inferred
- Status Bar: Inferred
- Top Bar: Inferred
- Bottom Bar: Inferred

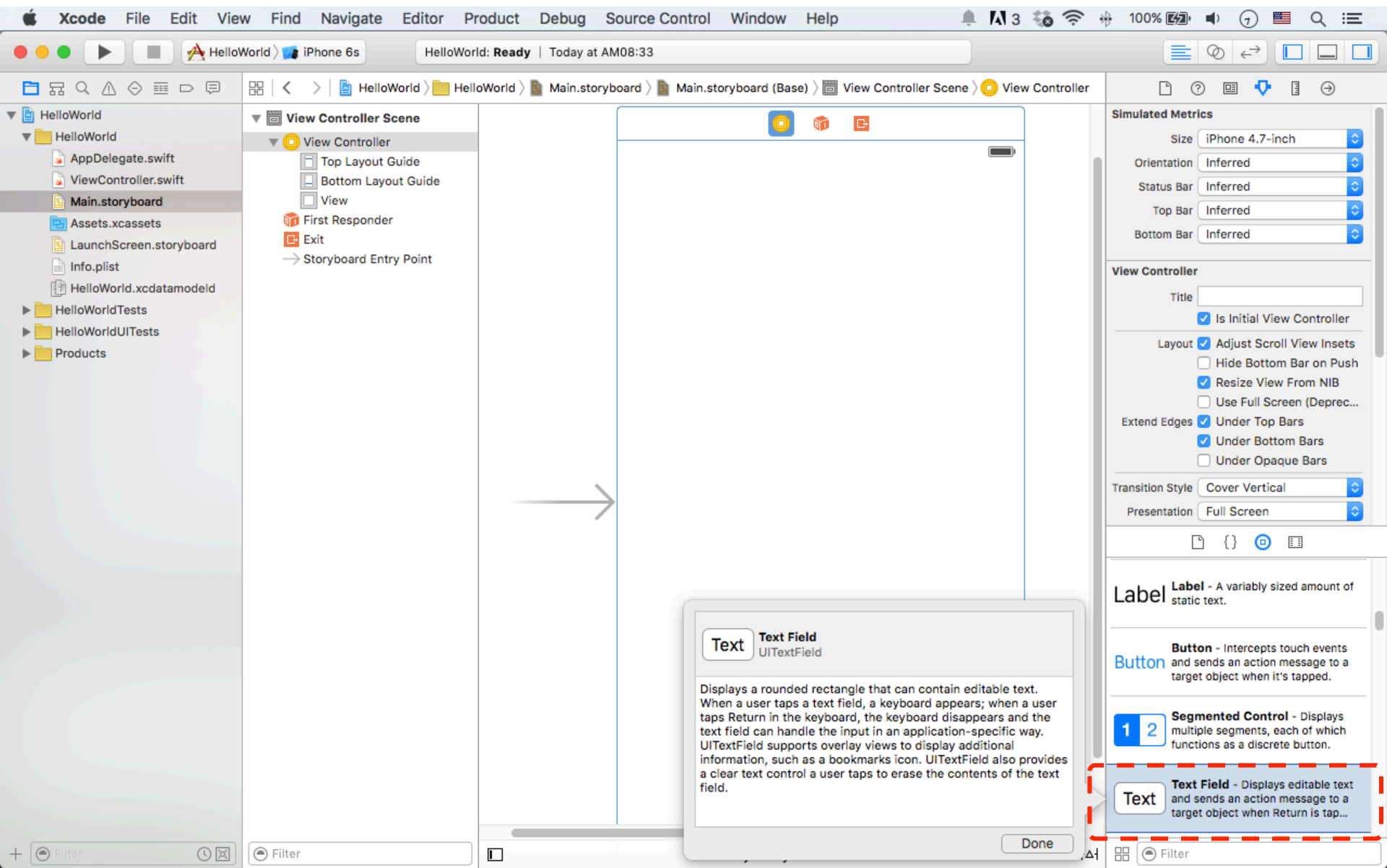
View Controller

- Title: []
- Is Initial View Controller
- Layout:
 - Adjust Scroll View Insets
 - Hide Bottom Bar on Push
 - Resize View From NIB
 - Use Full Screen (Deprec...)
- Extend Edges:
 - Under Top Bars
 - Under Bottom Bars
 - Under Opaque Bars
- Transition Style: Cover Vertical
- Presentation: Full Screen

Library (Bottom Right):

- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text Field** - Displays editable text and sends an action message to a target object when Return is tap...





Text Field

The screenshot displays the Xcode IDE interface for a project named "HelloWorld" on an iPhone 6s. The main window shows a storyboard with a "Round Style Text Field" widget placed on a "View Controller" scene. A blue arrow points from the "Text Field" widget in the storyboard to the "Text Field" properties panel on the right. The properties panel includes settings for Text (Plain), Color (Default), Font (System 14.0), Alignment, Placeholder (Placeholder Text), Background (Background Image), Disabled (Disabled Background Image), Border Style (Rounded), Clear Button (Never appears), Min Font Size (17), Capitalization (None), Correction (Default), and Spelling Checking (Default). Below the properties panel, there is a "Text Field" section with a description: "Text Field - Displays editable text and sends an action message to a target object when Return is tap...".

Text Field

Text Plain

Color Default

Font System 14.0

Alignment

Placeholder Placeholder Text

Background Background Image

Disabled Disabled Background Image

Border Style

Clear Button Never appears

Min Font Size 17

Adjust to Fit

Capitalization None

Correction Default

Spelling Checking Default

Label Label - A variably sized amount of static text.

Button Button - Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a target object when Return is tap...

File Explorer (Left):

- ▼ HelloWorld
 - ▼ HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
 - ▼ HelloWorldTests
 - ▼ HelloWorldUITests
 - ▼ Products

View Controller Scene (Middle-Left):

- ▼ View Controller Scene
 - ▼ View Controller
 - Top Layout Guide
 - Bottom Layout Guide
 - View
 - F Round Style Text...
 - First Responder
 - Exit
 - Storyboard Entry Point

Storyboard Canvas (Center):

Canvas Title: View Controller Scene > View > F Round Style Text Field

Canvas Content: A rounded rectangle text field with a placeholder.

Canvas Dimensions: wAny hAny

Text Field Inspector (Right):

Text Field

- Text: Plain
- Color: Default
- Font: System 14.0
- Alignment: [Left, Center, Right, Justified]
- Placeholder: Placeholder Text
- Background: Background Image
- Disabled: Disabled Background Image
- Border Style: [None, Solid, Recessed, Rounded]
- Clear Button: Never appears
- Min Font Size: 17
- Adjust to Fit:
- Capitalization: None
- Correction: Default
- Spelling Checking: Default

Label - A variably sized amount of static text.

Button - Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

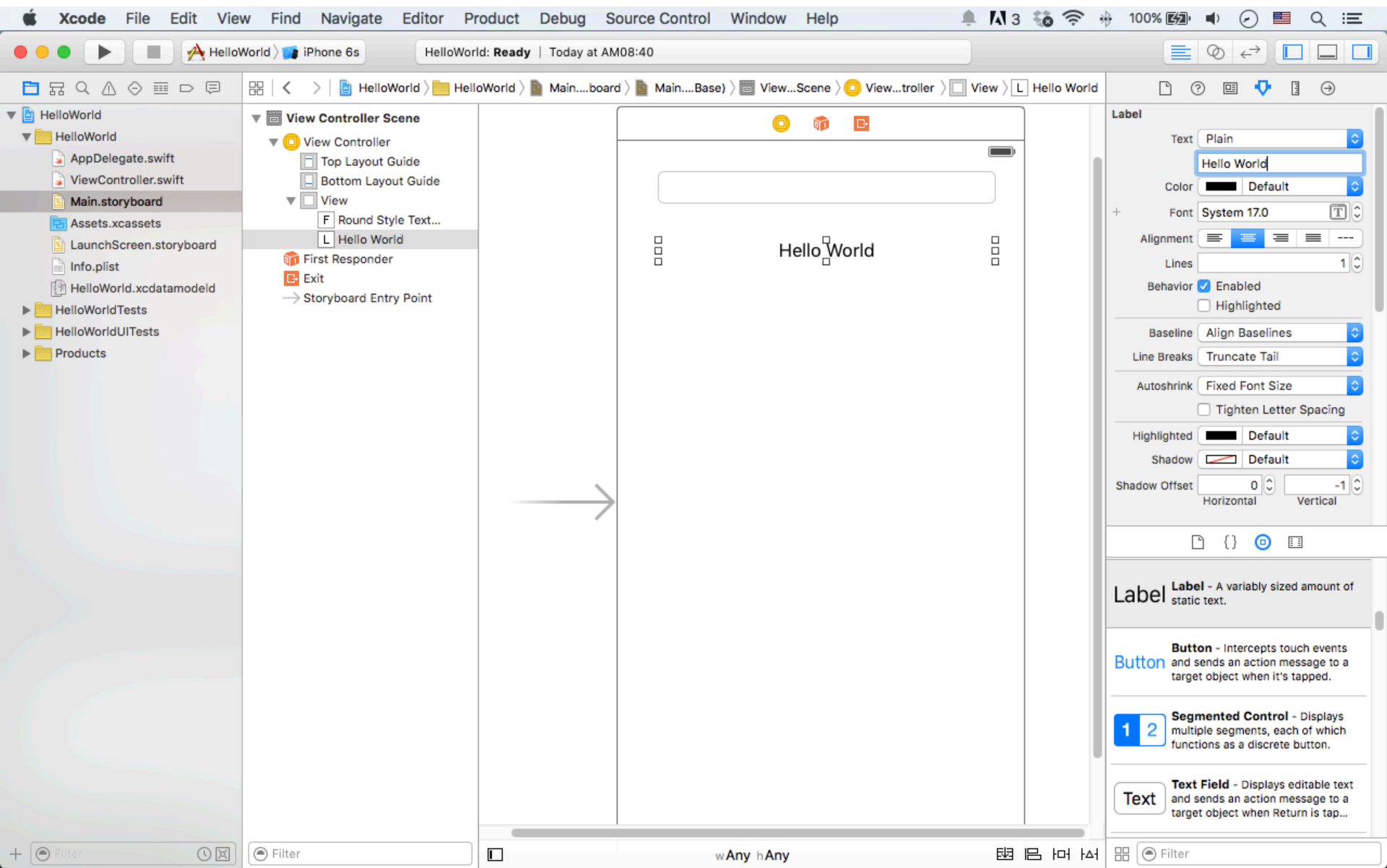
Text Field - Displays editable text and sends an action message to a target object when Return is tap...

Label

The screenshot displays the Xcode IDE with a storyboard for an iPhone 6s. The storyboard shows a 'View Controller Scene' containing a 'View Controller' with a 'View' and a 'Label' widget. The 'Label' widget is highlighted with a blue arrow pointing to its properties panel on the right. The properties panel includes fields for Text, Color, Font, Alignment, Lines, Behavior, Baseline, Line Breaks, Autoshrink, Highlighted, Shadow, and Shadow Offset. Below the properties panel, there is a section titled 'Label' with a description: 'Label - A variably sized amount of static text.' Below this, there are sections for 'Button', 'Segmented Control', and 'Text Field', each with a description of their functionality.

Xcode Interface Elements:

- Top Bar: Xcode, File, Edit, View, Find, Navigate, Editor, Product, Debug, Source Control, Window, Help. Status bar shows 100% zoom and AM08:37.
- Left Panel: Project Navigator showing 'HelloWorld' project with files like AppDelegate.swift, ViewController.swift, Main.storyboard, Assets.xcassets, LaunchScreen.storyboard, Info.plist, HelloWorld.xcdatamodeld, HelloWorldTests, HelloWorldUITests, and Products.
- Middle Panel: Document Navigator showing 'View Controller Scene' with 'View Controller' containing 'Top Layout Guide', 'Bottom Layout Guide', 'View', 'Round Style Text...', and 'Label'. Below it are 'First Responder' and 'Exit' options.
- Right Panel: Properties Inspector for 'Label' with settings for Text, Color, Font, Alignment, Lines, Behavior, Baseline, Line Breaks, Autoshrink, Highlighted, Shadow, and Shadow Offset.
- Bottom Panel: Size Inspector showing 'wAny hAny' constraints.



Project Navigator showing the file structure of the HelloWorld project:

- HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
 - HelloWorldTests
 - HelloWorldUITests
 - Products

Storyboard Editor showing the hierarchy and a preview of the UI:

- View Controller Scene
 - View Controller
 - Top Layout Guide
 - Bottom Layout Guide
 - View
 - Round Style Text... (F)
 - Hello World (L)
 - First Responder
 - Exit
 - Storyboard Entry Point

The preview window shows a white background with a text label that says "Hello World" in orange.

Inspector window for the selected Label:

- Label**
- Text: Plain (dropdown)
- Text: Hello World (input field)
- Color: [Orange color swatch]
- Font: System 24.0 (font size dropdown)
- Alignment: [Left, Center, Right, Justified, Default] (radio buttons)
- Lines: 1 (text field)
- Behavior: Enabled, Highlighted

Inspector window for the selected Label (right side):

- Label**
- Text: Plain (dropdown)
- Text: Hello World (input field)
- Color: [Orange color swatch]
- Font: System 24.0 (font size dropdown)
- Alignment: [Left, Center, Right, Justified, Default] (radio buttons)
- Lines: 1 (text field)
- Behavior: Enabled, Highlighted
- Baseline: Align Baselines (dropdown)
- Line Breaks: Truncate Tail (dropdown)
- Autoshrink: Fixed Font Size (dropdown)
- Tighten Letter Spacing
- Highlighted: [Black swatch] Default (dropdown)
- Shadow: [Default swatch] Default (dropdown)
- Shadow Offset: 0 (Horizontal), -1 (Vertical)

Library window showing UI components:

- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text** - Text Field - Displays editable text and sends an action message to a target object when Return is tap...

Button

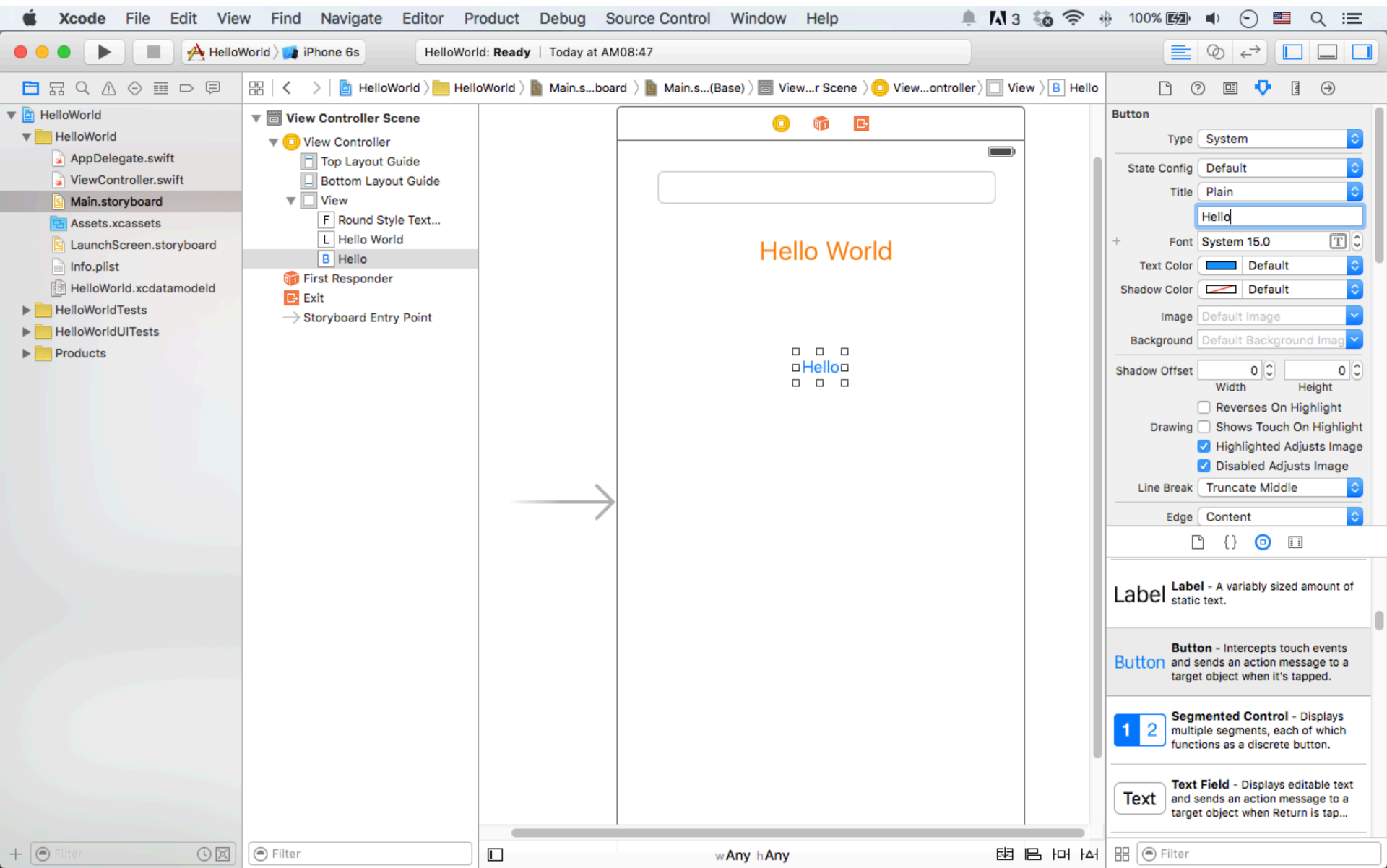
The screenshot displays the Xcode interface for editing a storyboard. The central canvas shows a mobile device frame with a white text field at the top and the text "Hello World" in orange below it. A blue button is positioned below the text, with a blue arrow pointing from the right-hand inspector to it. The left-hand sidebar shows a project tree with "Main.storyboard" selected. The middle-left sidebar shows a hierarchy of "View Controller Scene" > "View Controller" > "View" > "Button". The right-hand sidebar is the "Properties" inspector for the selected "Button", showing various configuration options.

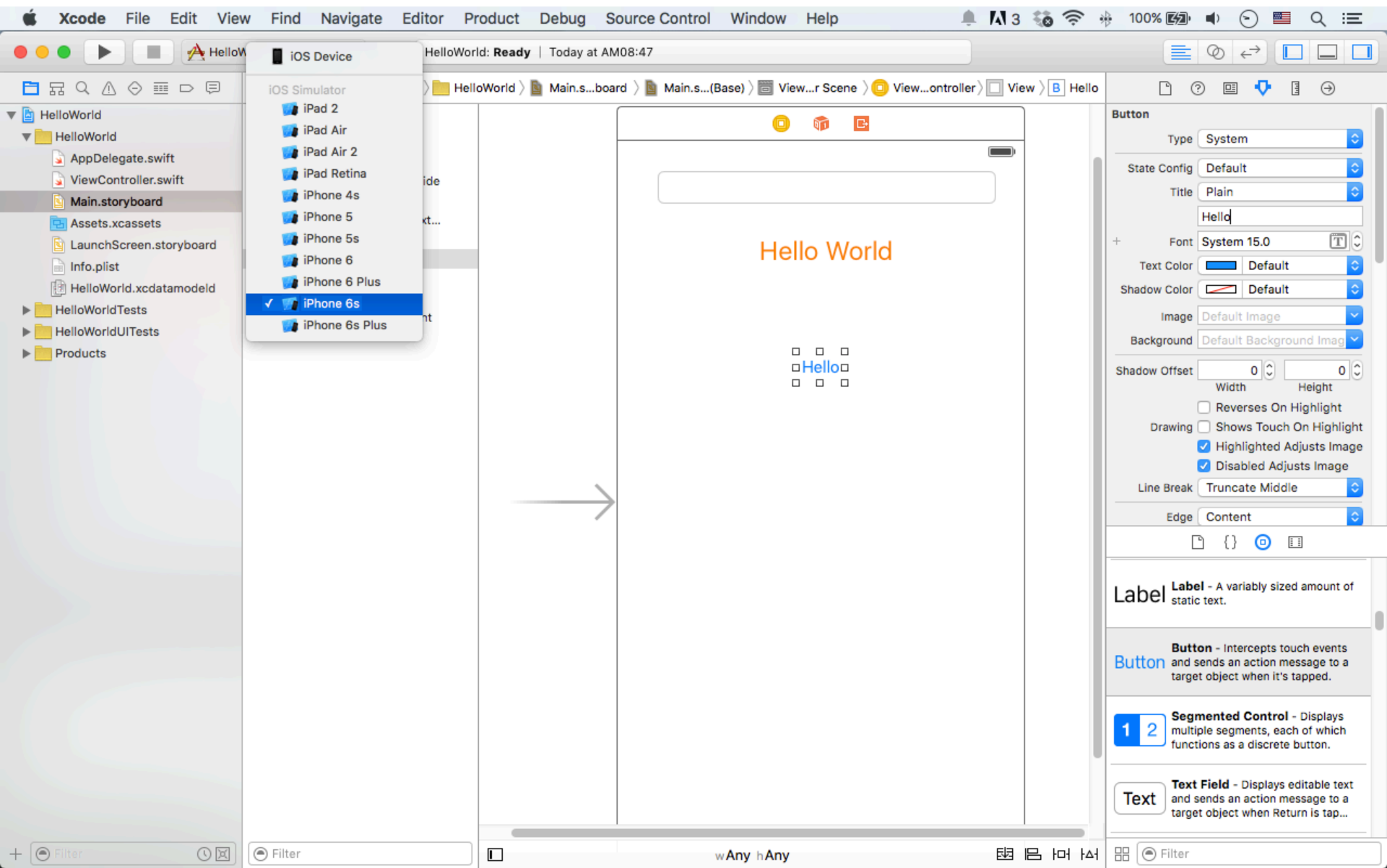
Button Properties:

- Type: System
- State Config: Default
- Title: Plain
- Button
- Font: System 15.0
- Text Color: Default
- Shadow Color: Default
- Image: Default Image
- Background: Default Background Image
- Shadow Offset: 0 (Width), 0 (Height)
- Reverses On Highlight
- Shows Touch On Highlight
- Highlighted Adjusts Image
- Disabled Adjusts Image
- Line Break: Truncate Middle
- Edge: Content

Component Descriptions:

- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text** - Text Field - Displays editable text and sends an action message to a target object when Return is tap...





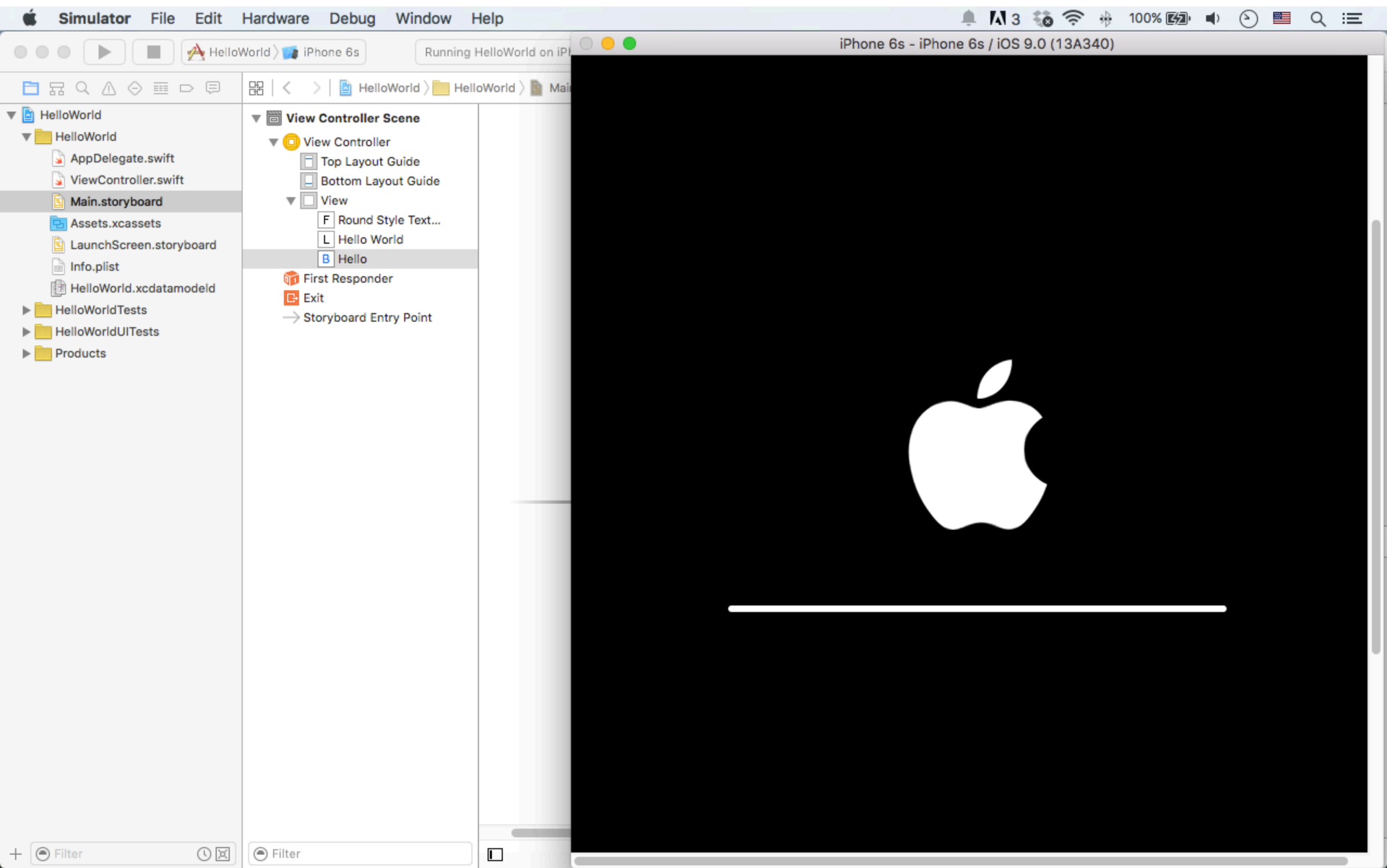
Build and Run

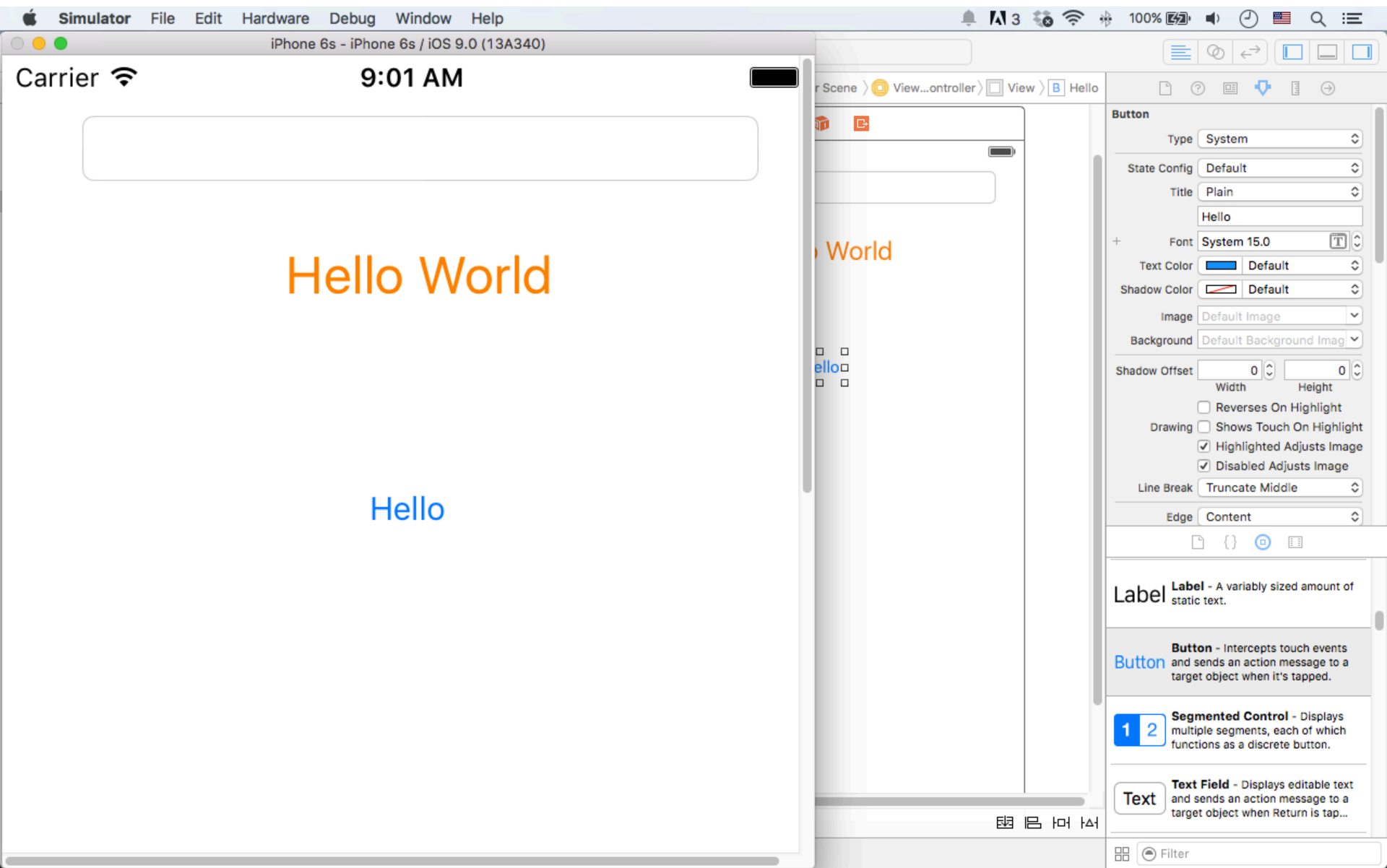
The screenshot displays the Xcode IDE interface. At the top, the menu bar includes 'Xcode', 'File', 'Edit', 'View', 'Find', 'Navigate', 'Editor', 'Product', 'Debug', 'Source Control', 'Window', and 'Help'. The status bar shows 'HelloWorld: Ready' and 'Today at AM08:48'. A red box highlights the play button (a right-pointing triangle) in the top-left corner of the toolbar.

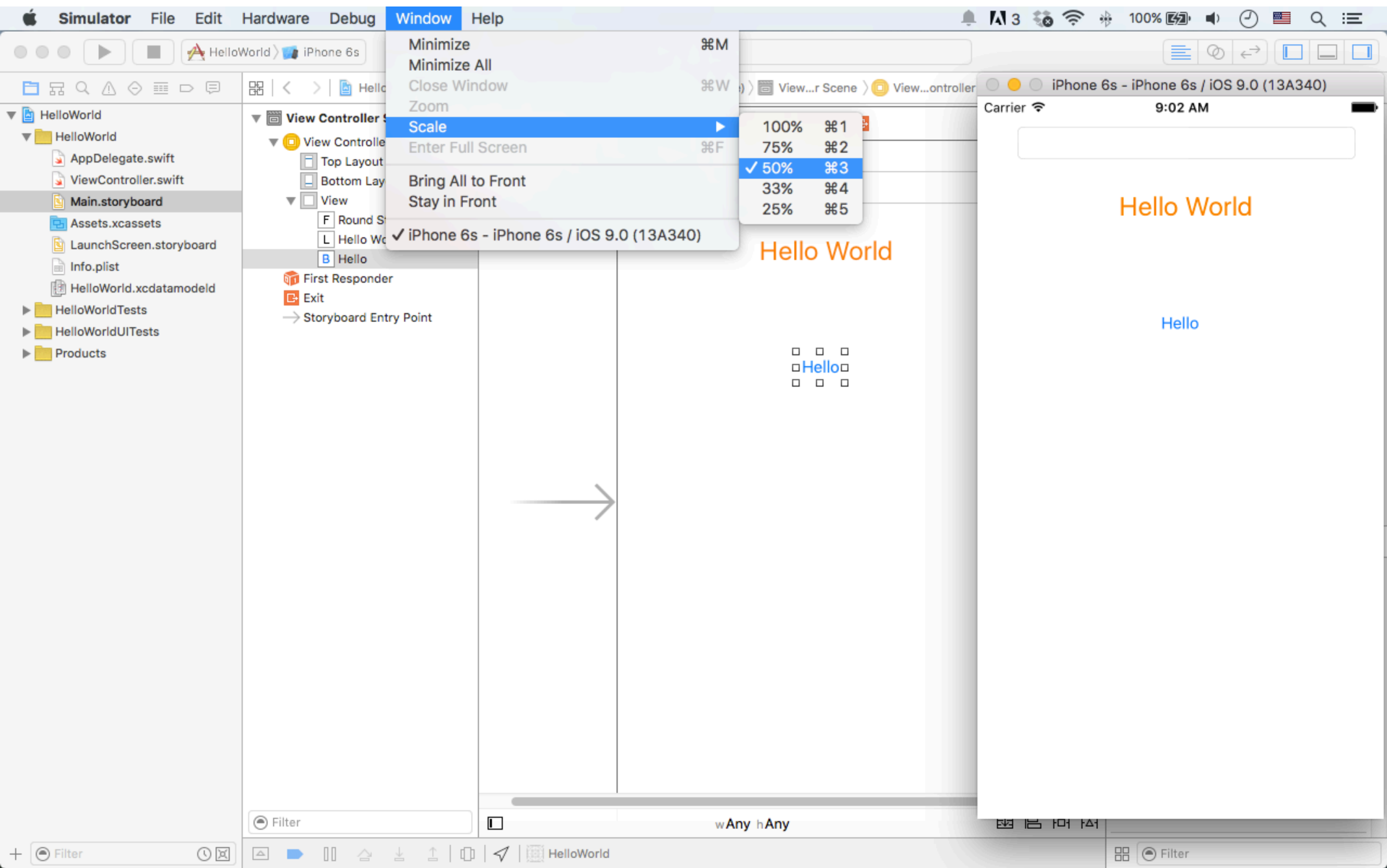
The main workspace is divided into three panes:

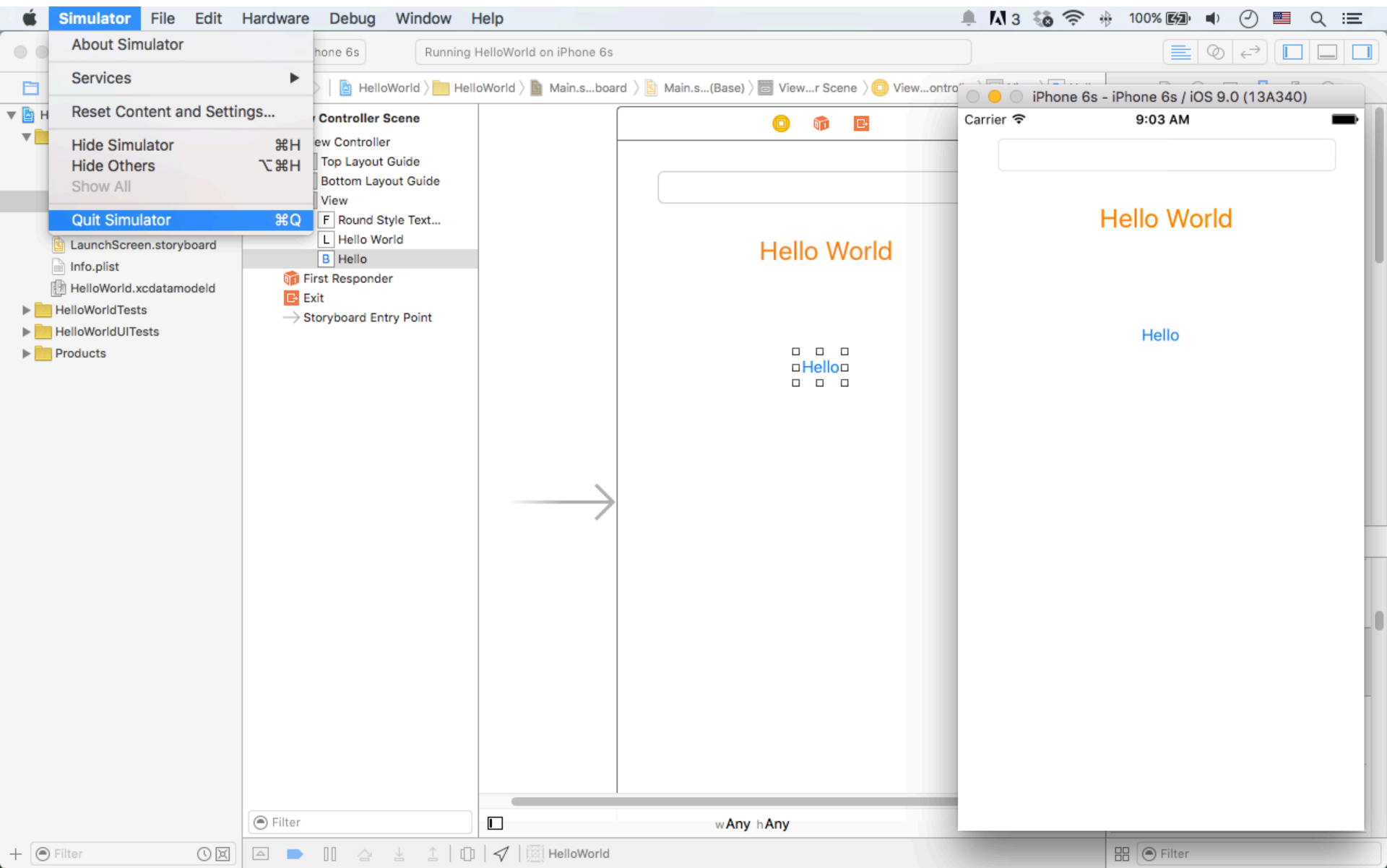
- Left Pane (Project Navigator):** Shows the project structure for 'HelloWorld'. The 'Main.storyboard' file is selected.
- Middle Pane (Storyboard):** Displays the 'View Controller Scene' with a 'View Controller' containing a 'View' with a 'Hello' label. A large grey arrow points from the storyboard to the simulator.
- Right Pane (Inspector and Library):** The 'Button' inspector is active, showing properties like 'Type', 'State Config', 'Title', 'Font', 'Text Color', 'Shadow Color', 'Image', 'Background', 'Shadow Offset', 'Reverses On Highlight', 'Shows Touch On Highlight', 'Highlighted Adjusts Image', 'Disabled Adjusts Image', 'Line Break', and 'Edge'. Below the inspector is a library of UI components including 'Label', 'Button', 'Segmented Control', and 'Text Field'.

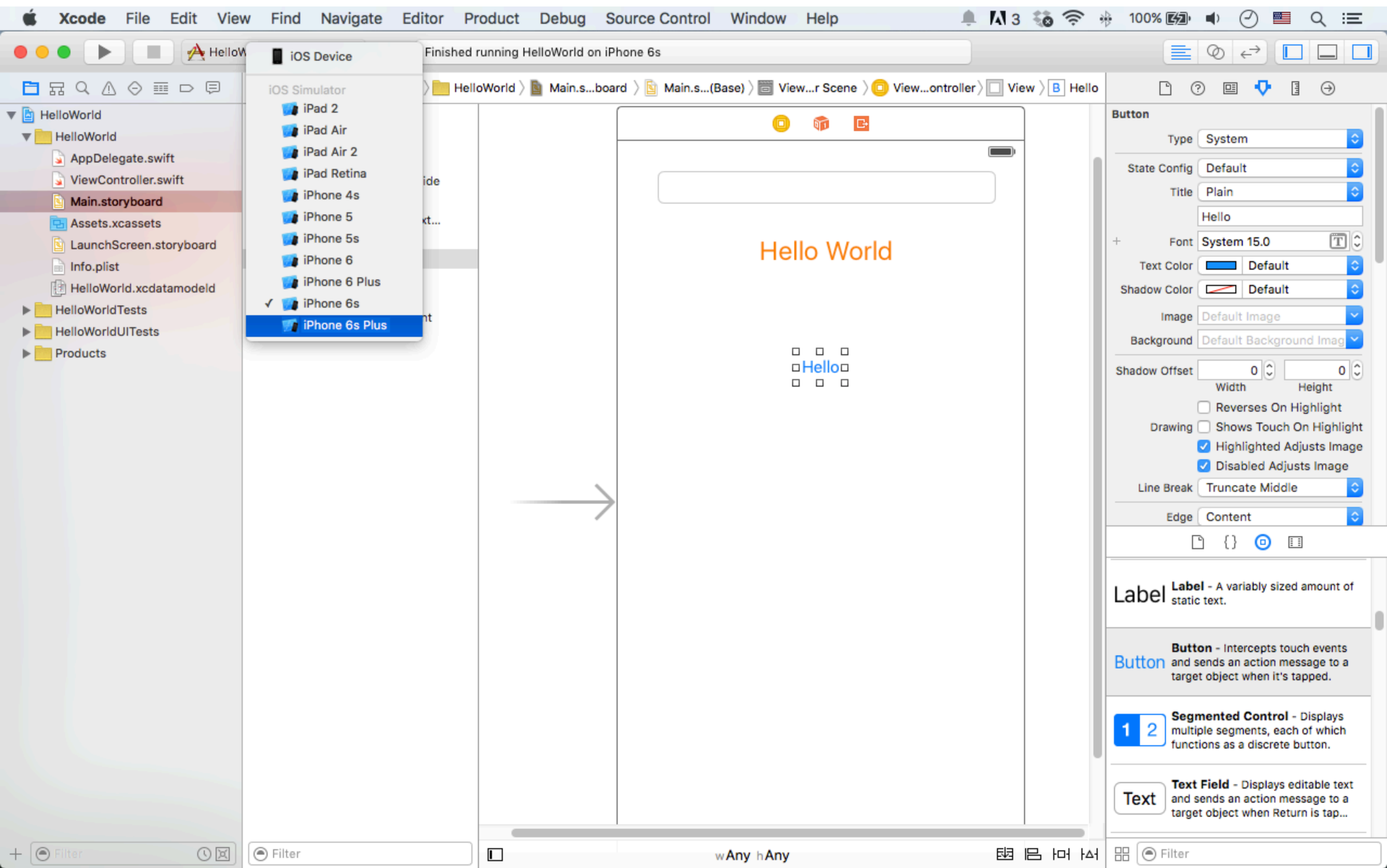
At the bottom, the status bar shows 'wAny hAny' and various tool icons.

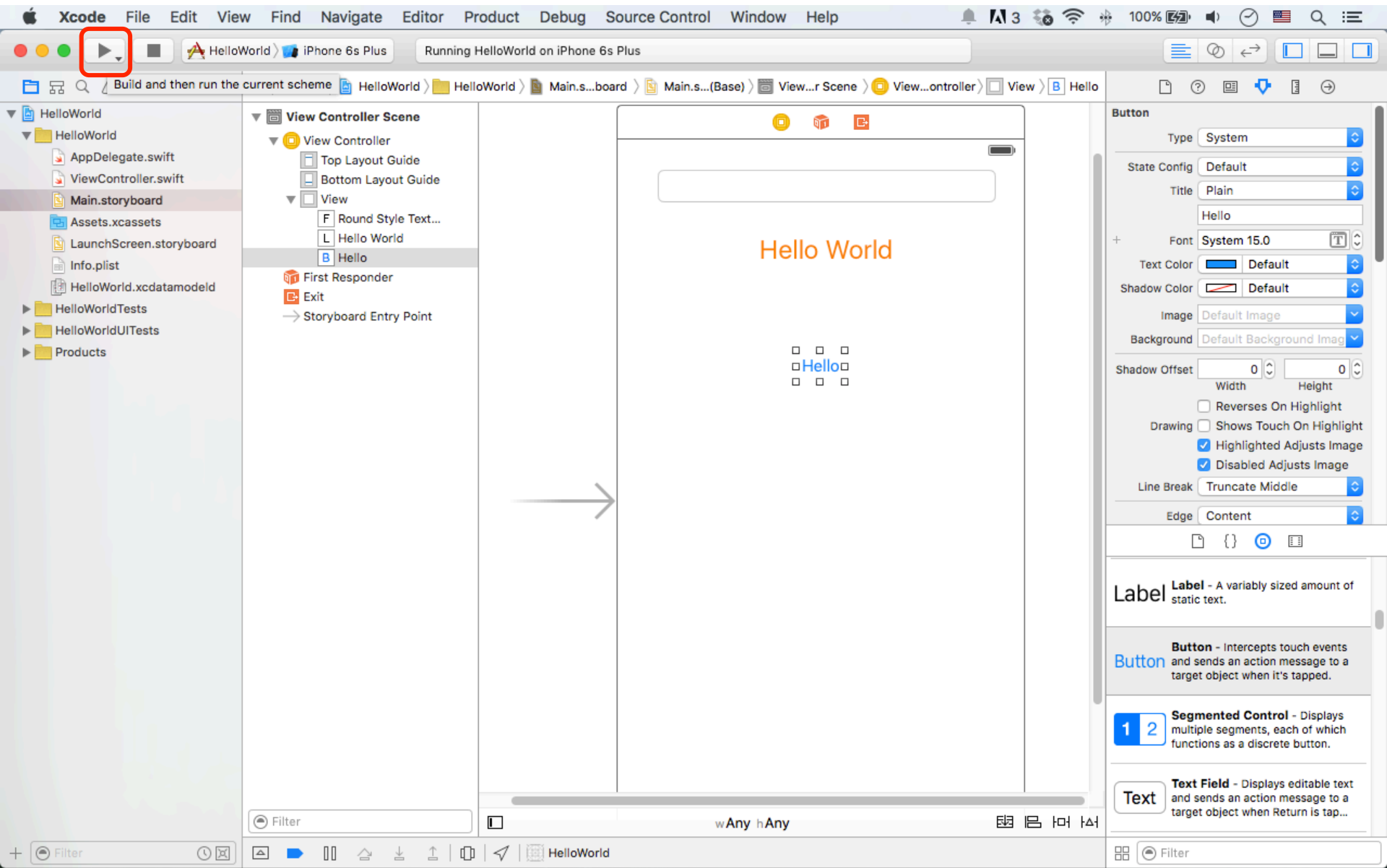














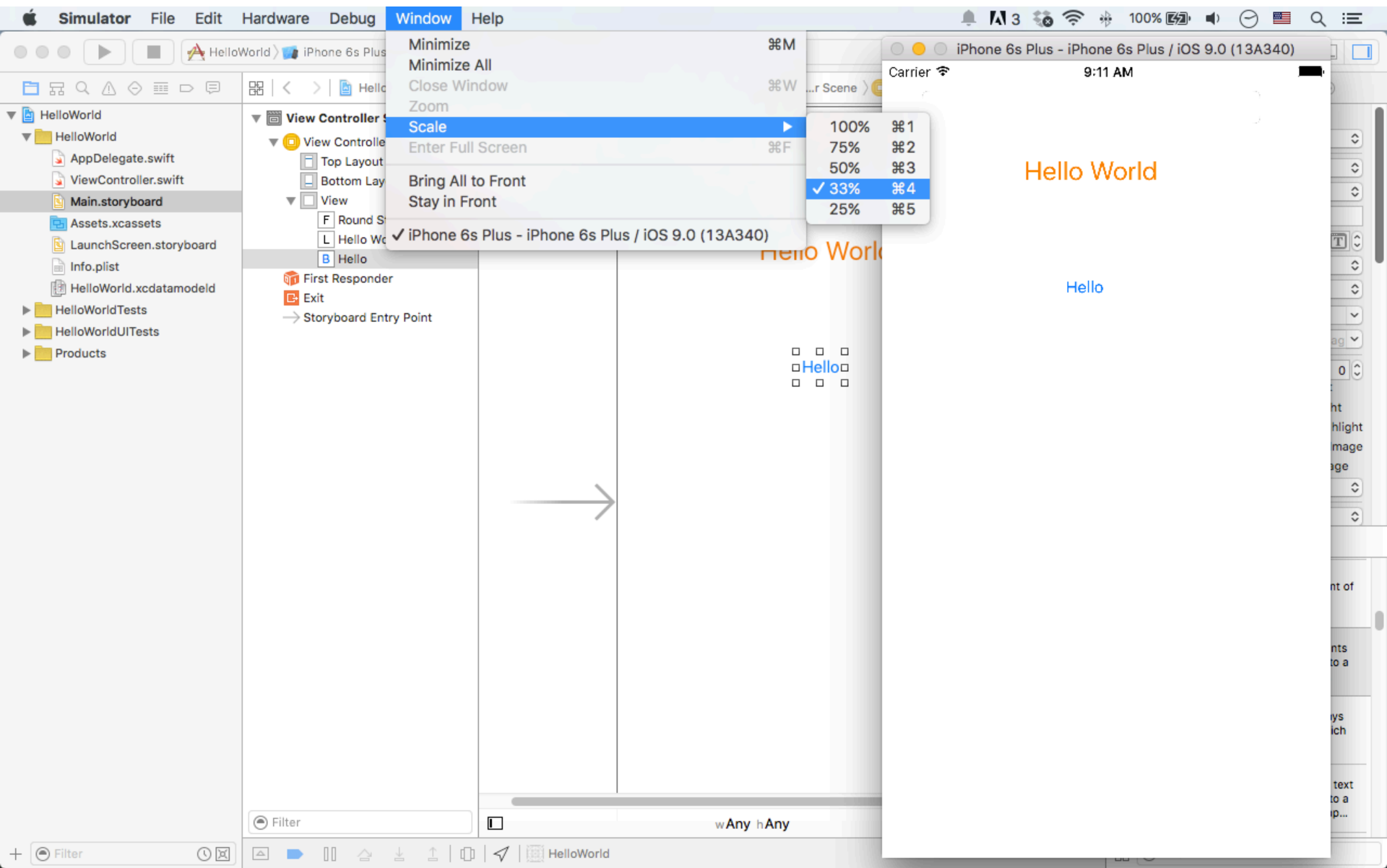
Carrier 

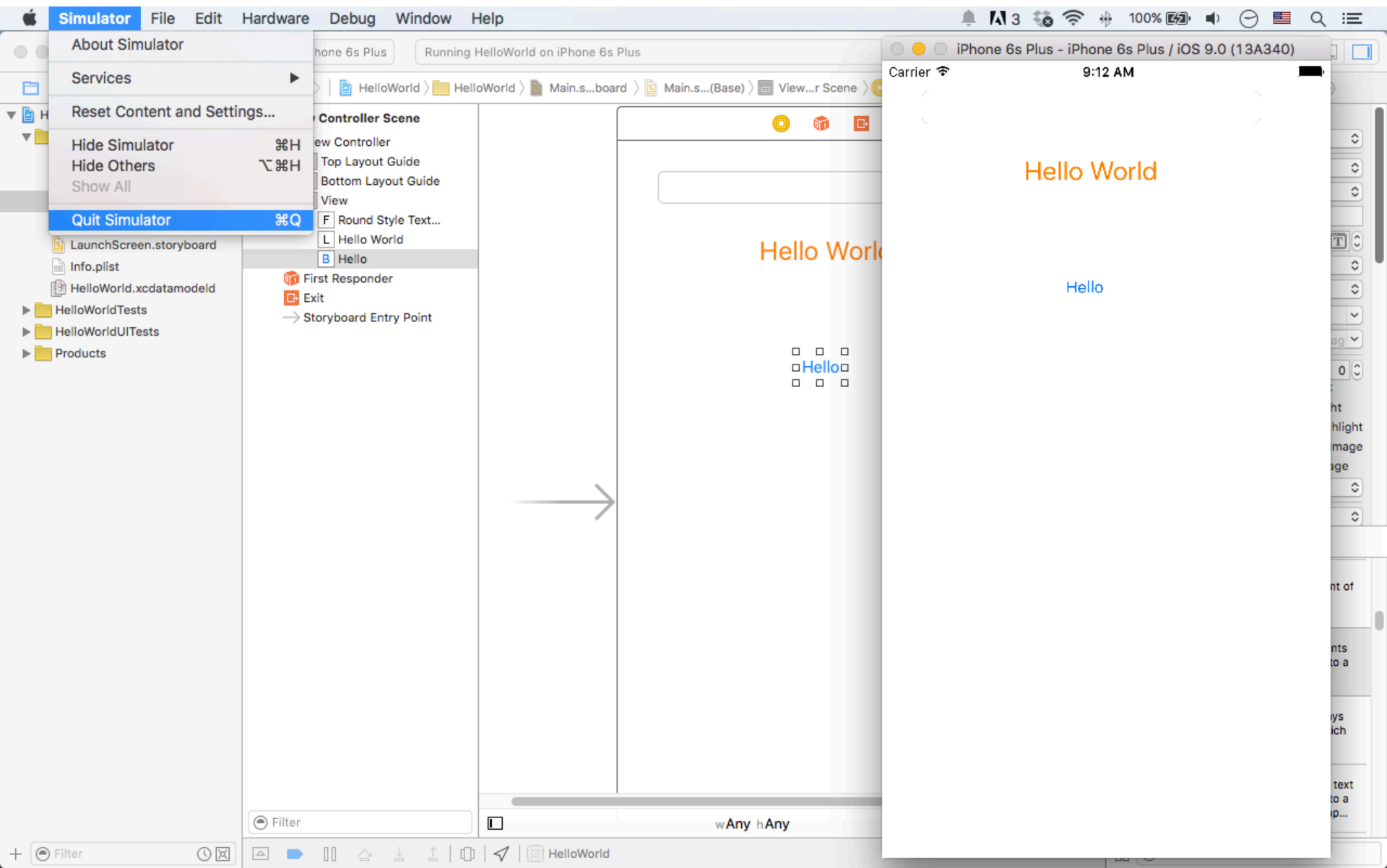
9:10 AM



Hello World

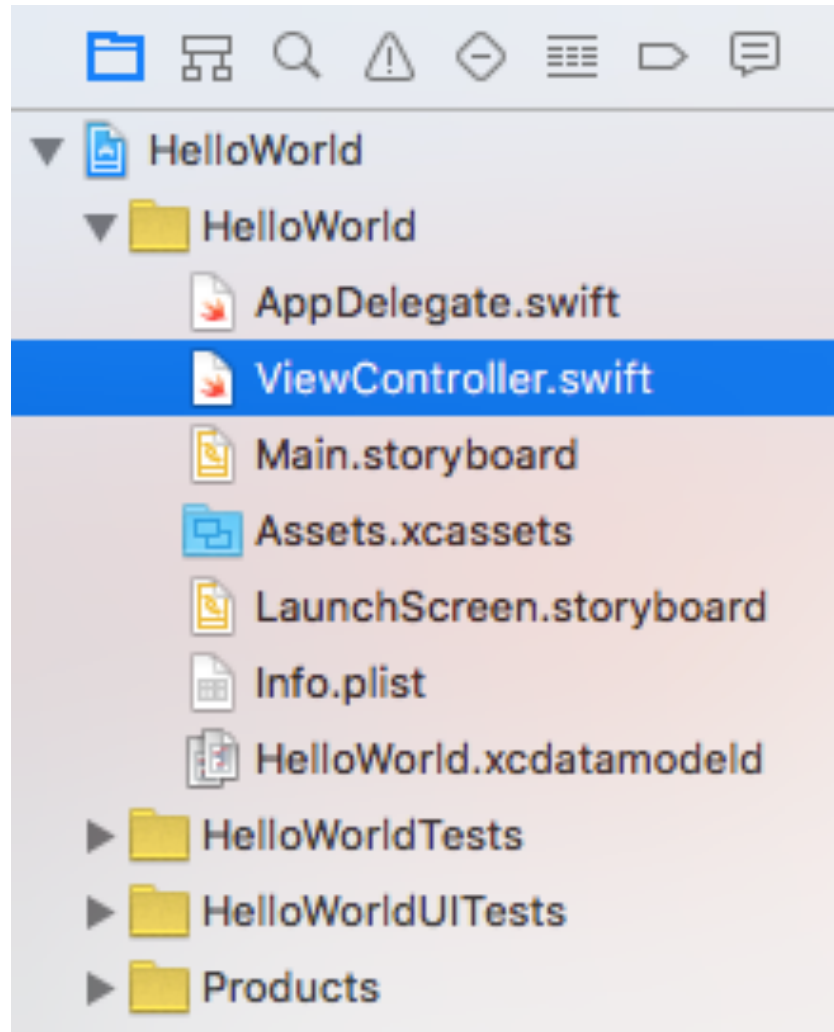
Hello





Main.storyboard (UI)

ViewController.swift (Code)



ViewController.swift (Code)

The screenshot shows the Xcode IDE interface. The top menu bar includes Xcode, File, Edit, View, Find, Navigate, Editor, Product, Debug, Source Control, Window, and Help. The status bar at the top right shows system icons and a 100% zoom level. The main editor window displays the code for ViewController.swift, which is part of a project named HelloWorld on an iPhone 6s. The code includes comments, an import statement for UIKit, and two overridden methods: viewDidLoad() and didReceiveMemoryWarning().

```
//  
// ViewController.swift  
// HelloWorld  
//  
// Created by iMyday on 10/7/15.  
// Copyright © 2015 imtku. All rights reserved.  
//  
import UIKit  
  
class ViewController: UIViewController {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any additional setup after loading the view, typically from a nib.  
    }  
  
    override func didReceiveMemoryWarning() {  
        super.didReceiveMemoryWarning()  
        // Dispose of any resources that can be recreated.  
    }  
  
}
```

The right sidebar contains a Quick Help section with the text "No Quick Help". Below this, there is a list of UI elements with their descriptions:

- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text** - Text Field - Displays editable text and sends an action message to a target object when Return is tap...

IBOutlet and IBAction

- IBOutlet
 - Interface Builder Outlet
- IBAction
 - Interface Builder Action

Xcode interface showing a project named "HelloWorld" for an iPhone 6s. The interface is divided into three main sections:

- Left Panel (Project Navigator):** Shows the project structure. The "Main.storyboard" file is selected and highlighted in blue. Other files include AppDelegate.swift, ViewController.swift, Assets.xcassets, LaunchScreen.storyboard, Info.plist, and HelloWorld.xcdatamodeld.
- Center Panel (Storyboard):** Displays a visual representation of the app's interface. It features a white background with a search bar at the top, the text "Hello World" in orange, and a blue button labeled "Hello" with a white shadow effect.
- Right Panel (Code Editor):** Shows the Swift code for the ViewController. The code includes comments, an import statement for UIKit, and the class definition for ViewController, which overrides viewDidLoad() and didReceiveMemoryWarning().

```
// ViewController.swift
// HelloWorld
//
// Created by iMyday on 10/7/15.
// Copyright © 2015 imtku. All rights reserved.
//

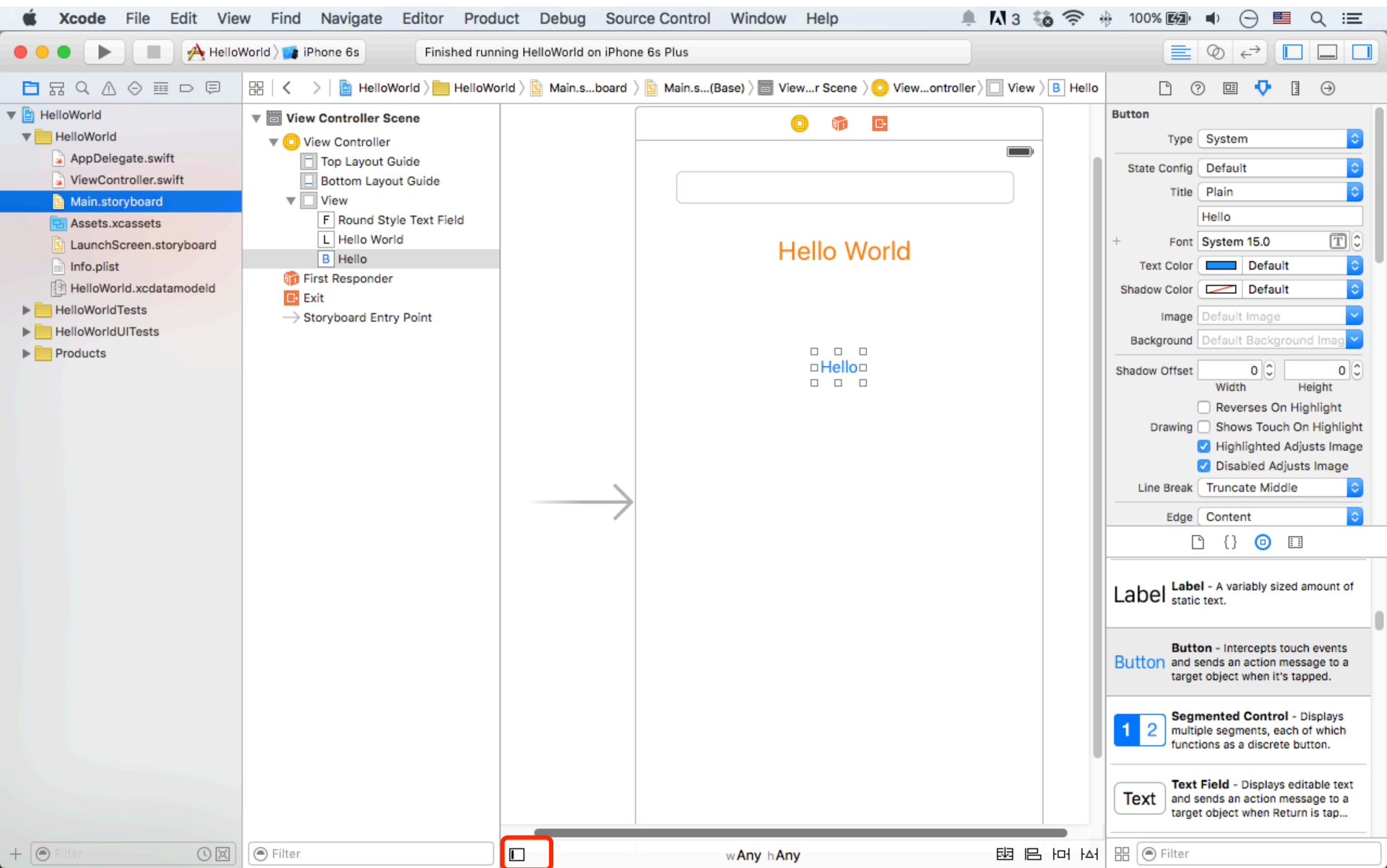
import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view,
        // typically from a nib.
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }

}
```



File Explorer showing project structure:

- HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - HelloWorld.xcdatamodeld
- HelloWorldTests
- HelloWorldUITests
- Products

Storyboard preview showing a text field at the top, the text "Hello World" in orange, and a "Hello" label with a grid below it. A large arrow points from the left sidebar towards the storyboard.

Navigation bar with icons for storyboard, settings, and other tools.

Inspector panel for a Button:

- Type: System
- State Config: Default
- Title: Plain
- Text: Hello
- Font: System 15.0
- Text Color: Default
- Shadow Color: Default
- Image: Default Image
- Background: Default Background Image
- Shadow Offset: Width 0, Height 0
- Reverses On Highlight:
- Shows Touch On Highlight:
- Highlighted Adjusts Image:
- Disabled Adjusts Image:
- Line Break: Truncate Middle
- Edge: Content

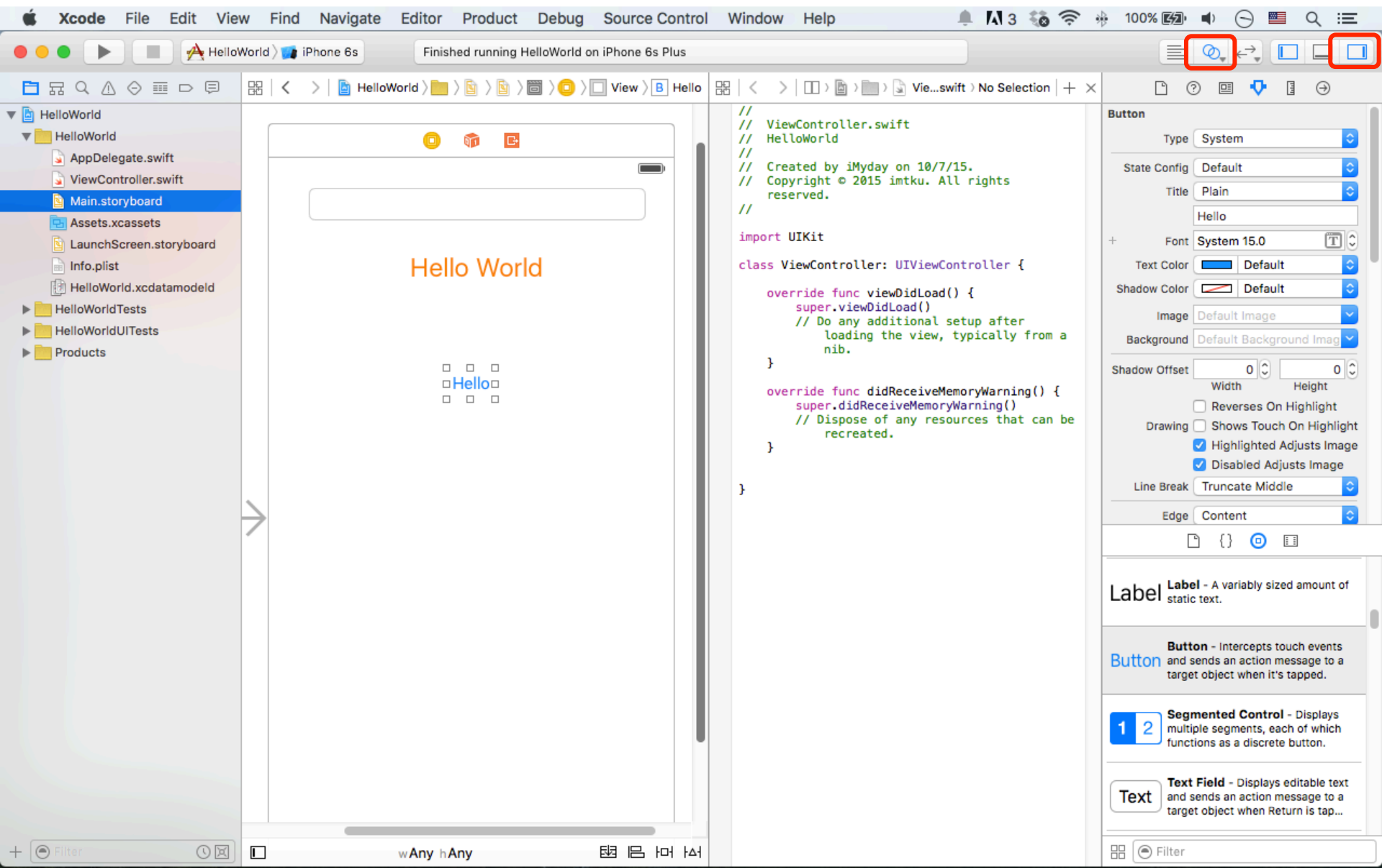
Library panel showing UI element descriptions:

- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text Field** - Displays editable text and sends an action message to a target object when Return is tap...

Bottom toolbar with icons for zooming and a red box around the 'Add New View' icon.

wAny hAny

Bottom toolbar with icons for zooming and a 'Filter' input field.



```
//  
// ViewController.swift  
// HelloWorld  
//  
// Created by iMyday on 10/7/15.  
// Copyright © 2015 imtku. All rights reserved.  
//  
import UIKit  
  
class ViewController: UIViewController {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any additional setup after  
        // loading the view, typically from a nib.  
    }  
  
    override func didReceiveMemoryWarning() {  
        super.didReceiveMemoryWarning()  
        // Dispose of any resources that can be  
        // recreated.  
    }  
}
```

Button

Type System

State Config Default

Title Plain

Hello

Font System 15.0

Text Color Default

Shadow Color Default

Image Default Image

Background Default Background Image

Shadow Offset 0 0

Width Height

Reverses On Highlight

Shows Touch On Highlight

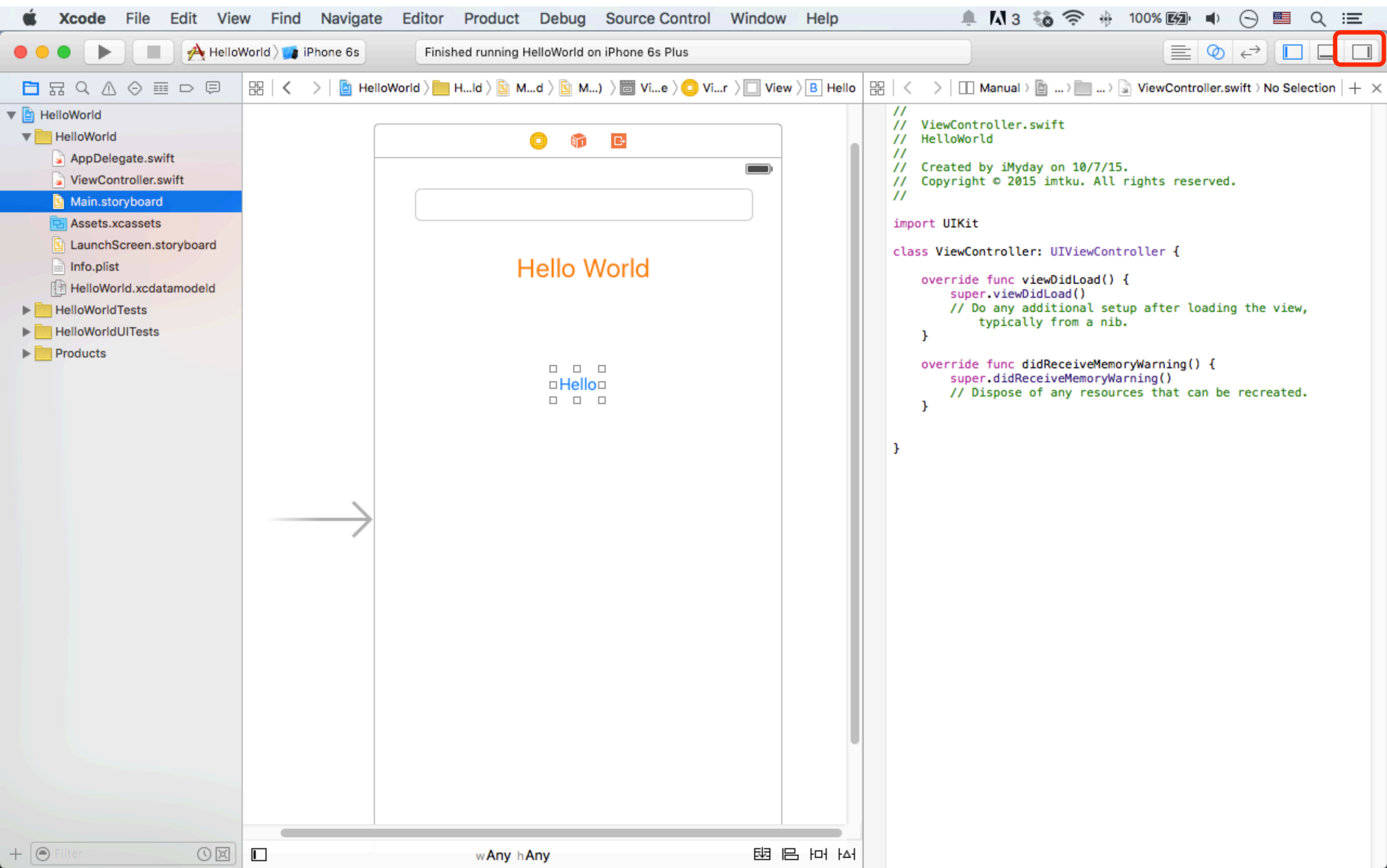
Highlighted Adjusts Image

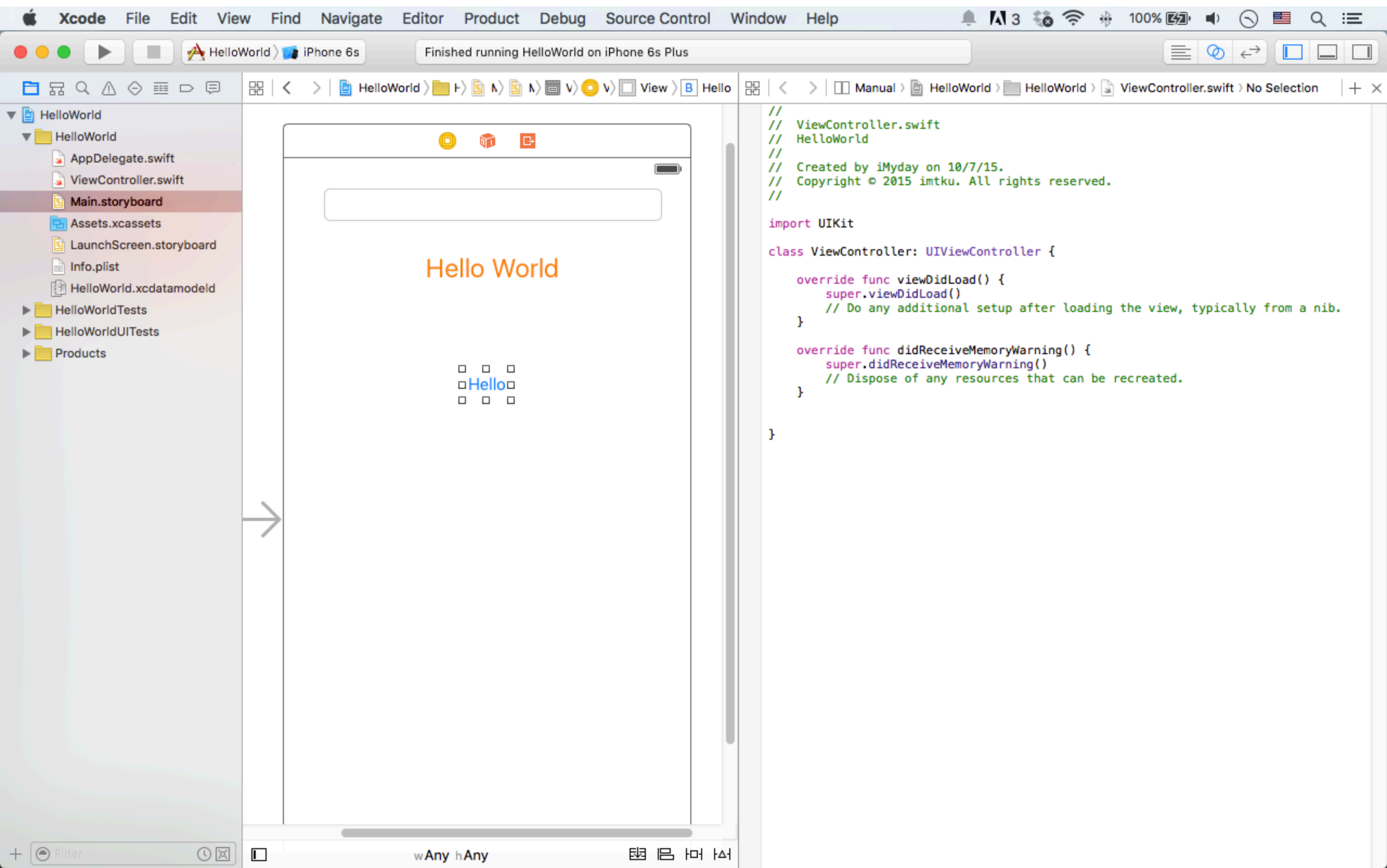
Disabled Adjusts Image

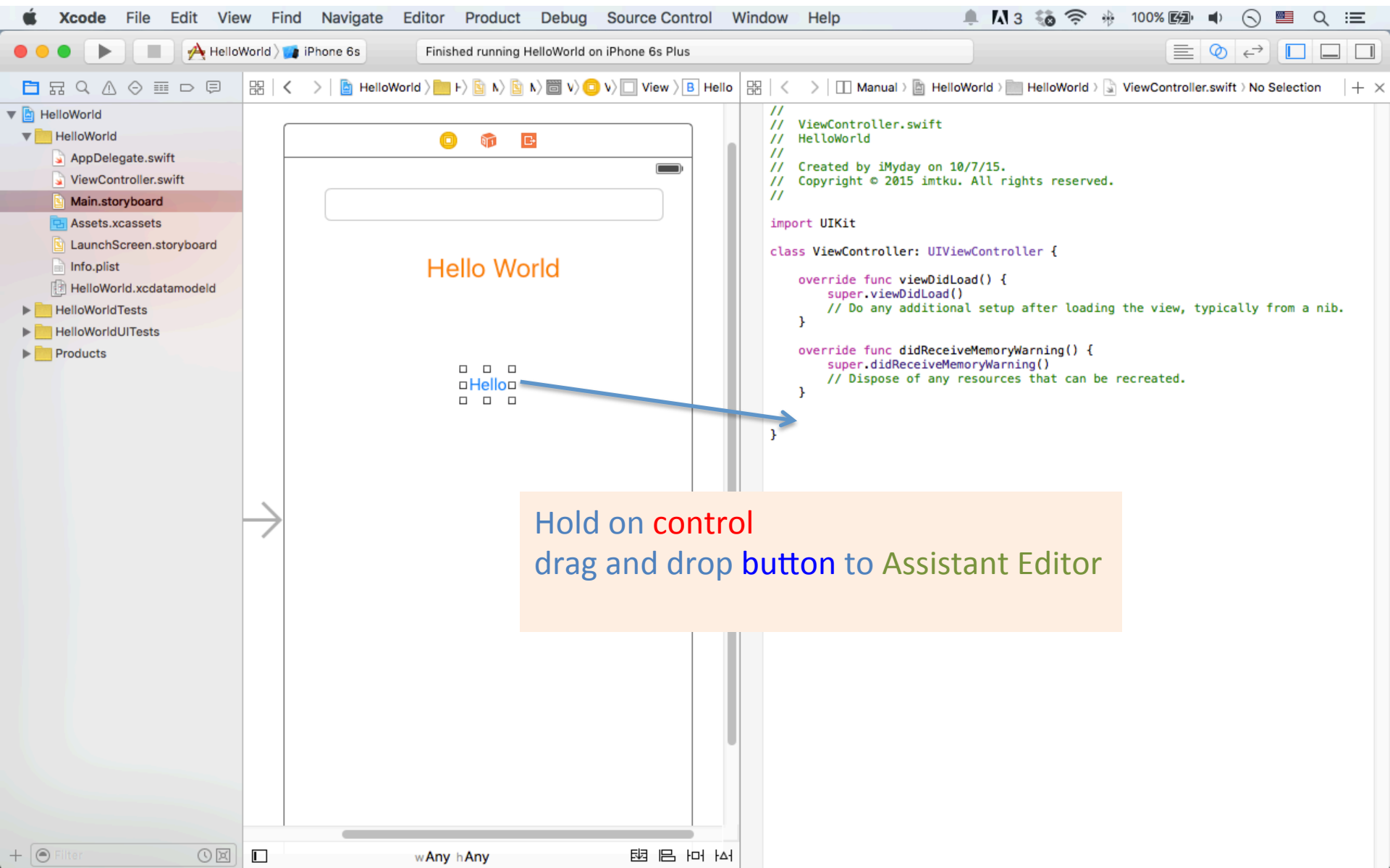
Line Break Truncate Middle

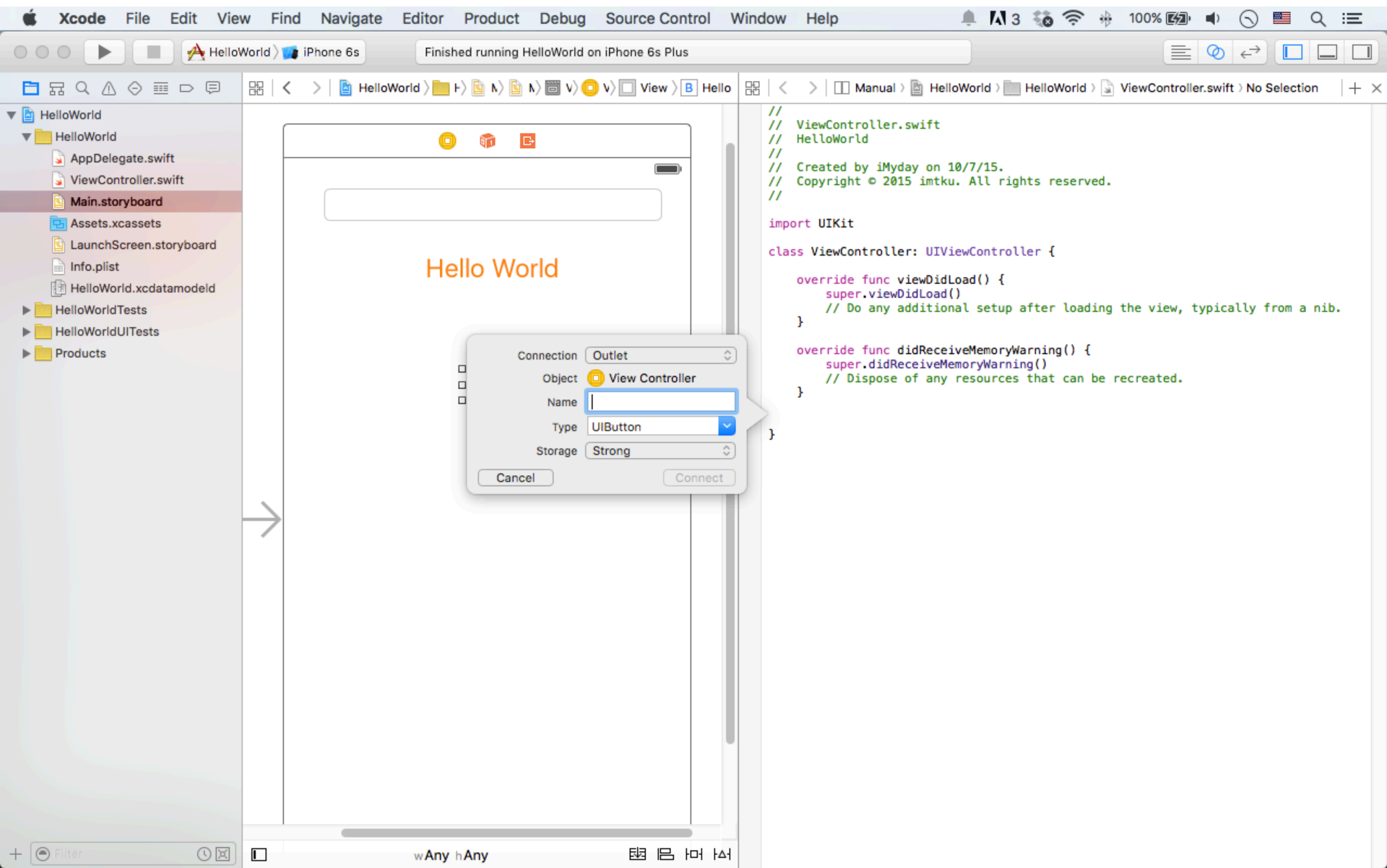
Edge Content

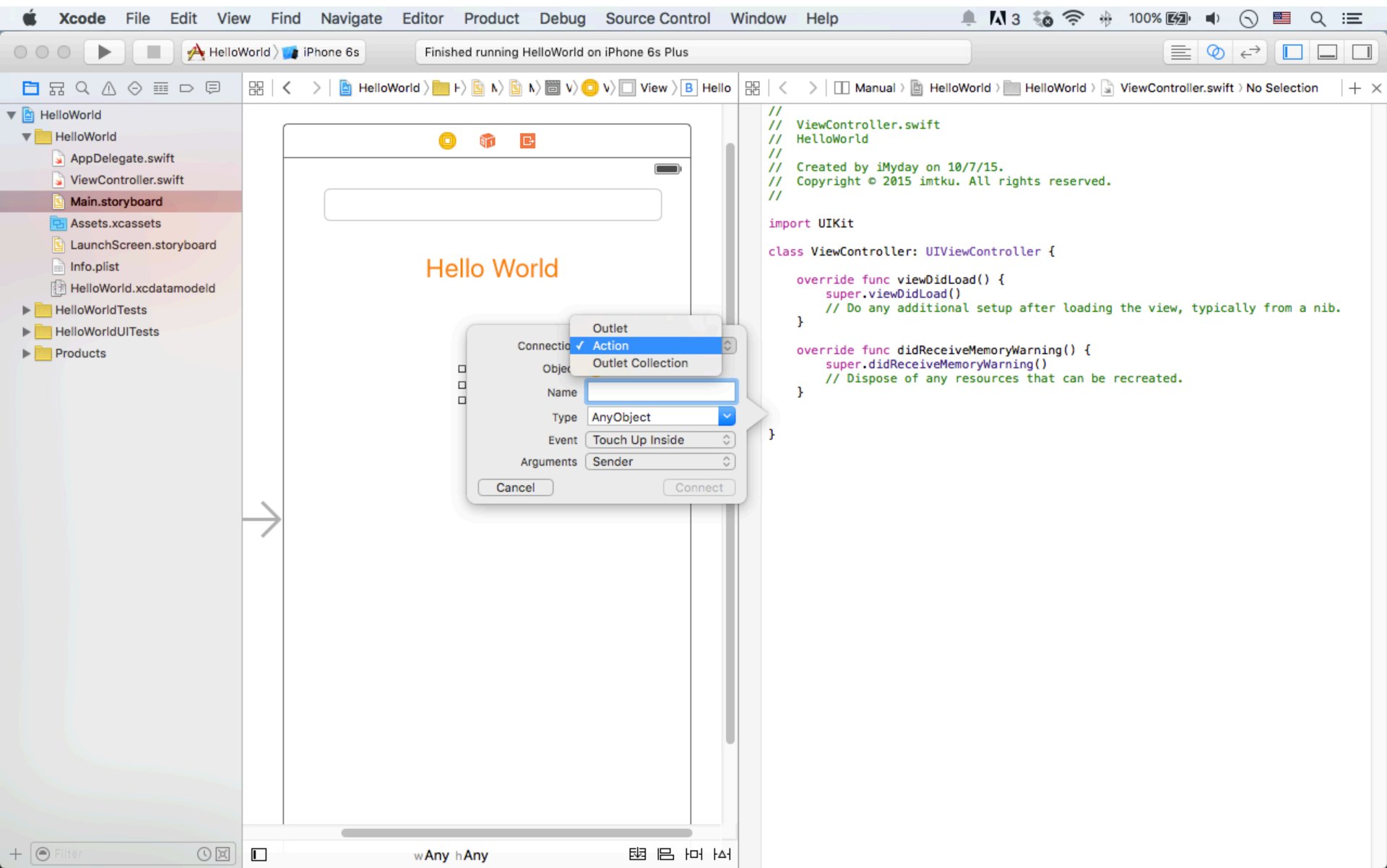
- Label** - A variably sized amount of static text.
- Button** - Intercepts touch events and sends an action message to a target object when it's tapped.
- 1 2 Segmented Control** - Displays multiple segments, each of which functions as a discrete button.
- Text** **Text Field** - Displays editable text and sends an action message to a target object when Return is tap...











File Explorer sidebar:

- HelloWorld
 - 2 targets, iOS SDK 8.0
 - HelloWorld
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Images.xcassets
 - LaunchScreen.xib
 - HelloWorld.xcdatamodeld
 - Supporting Files
 - HelloWorldTests
 - Products

Storyboard Editor showing a view with the text "Hello World". A context menu is open over a UI element, showing options for connection type: Outlet, Action, and Outlet Collection. The "Action" option is currently selected.

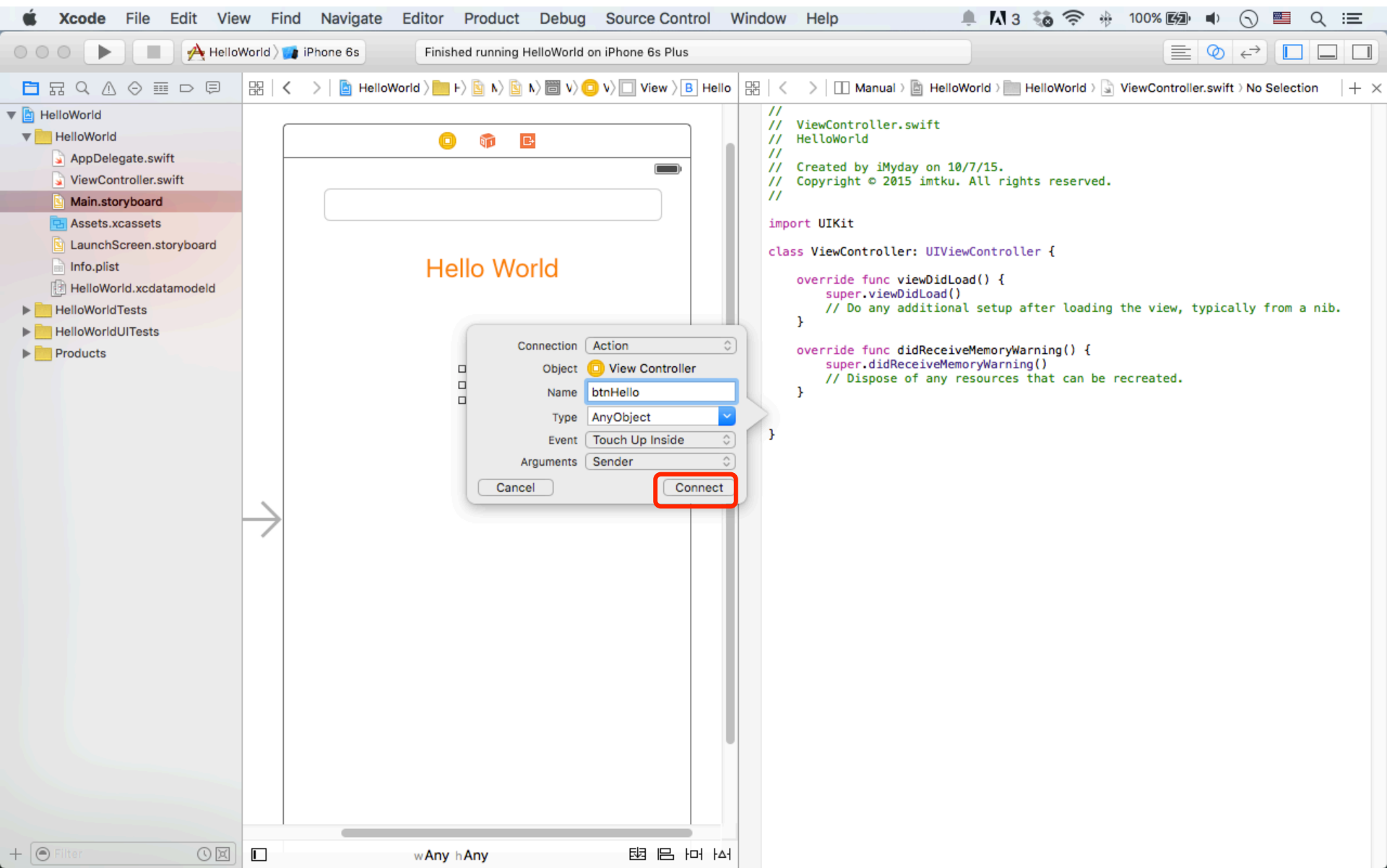
```
//
// ViewController.swift
// HelloWorld
//
// Created by iMyday on 2014/10/8.
// Copyright (c) 2014年 imtku. All rights reserved.

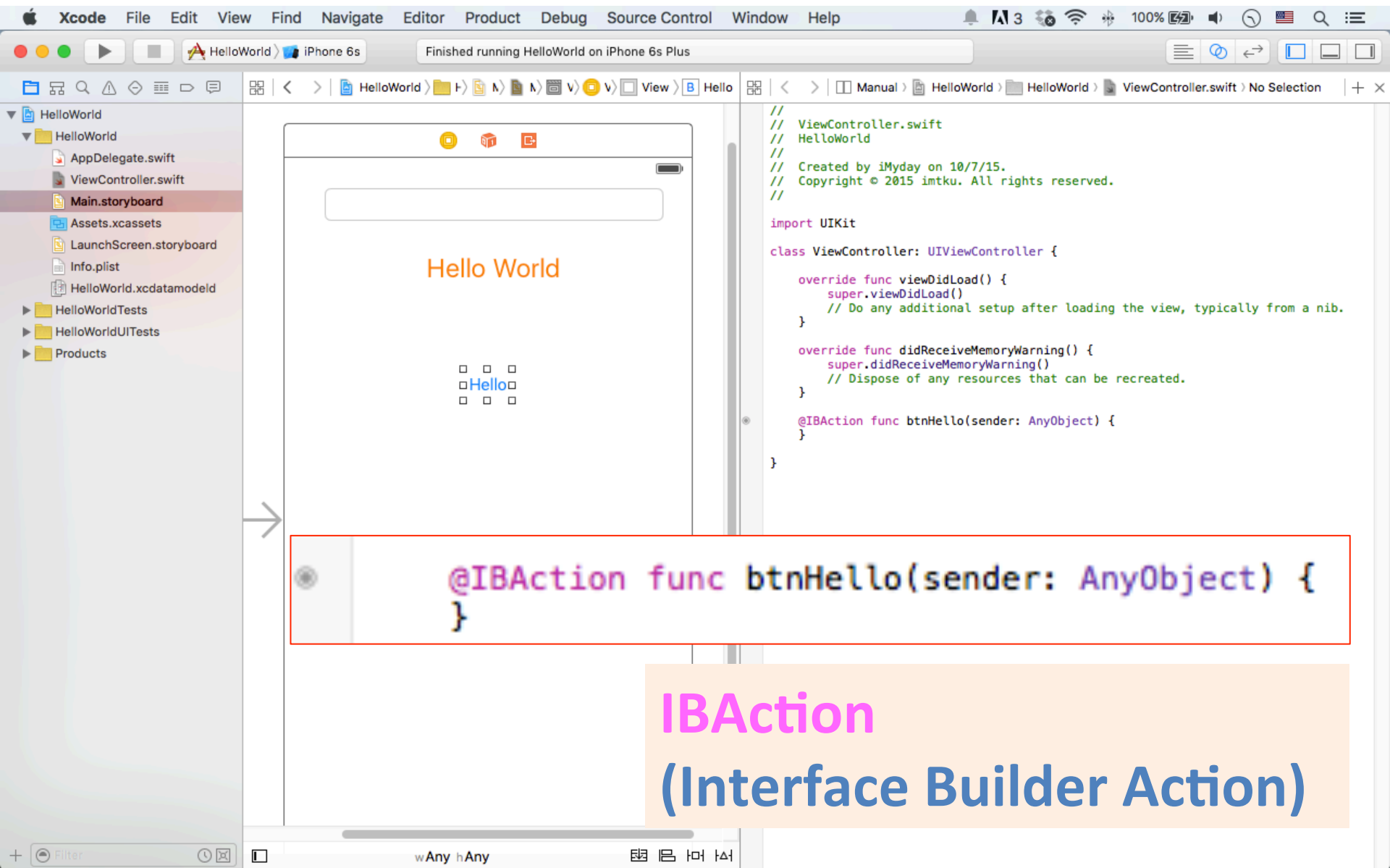
import UIKit

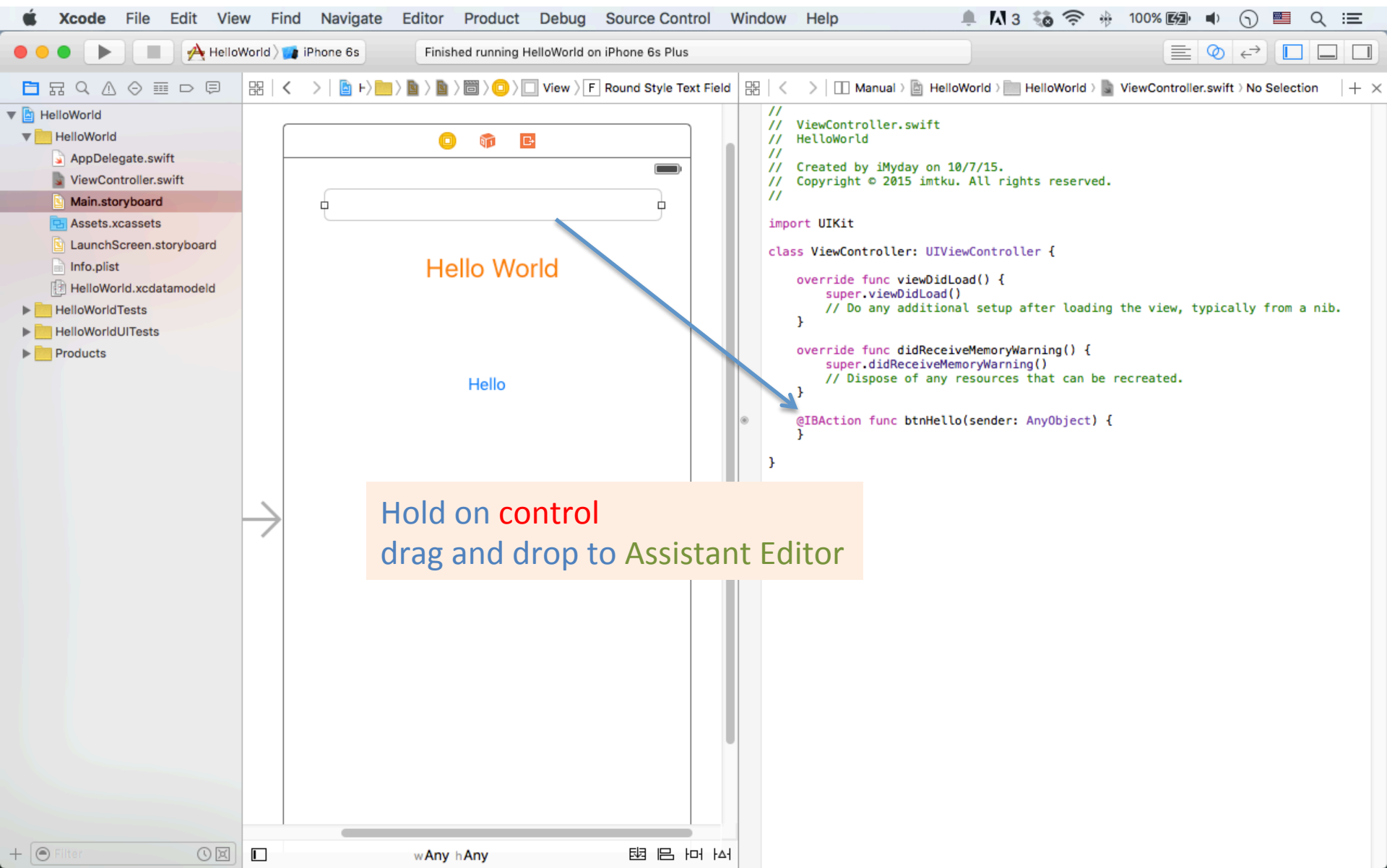
class ViewController: UIViewController {

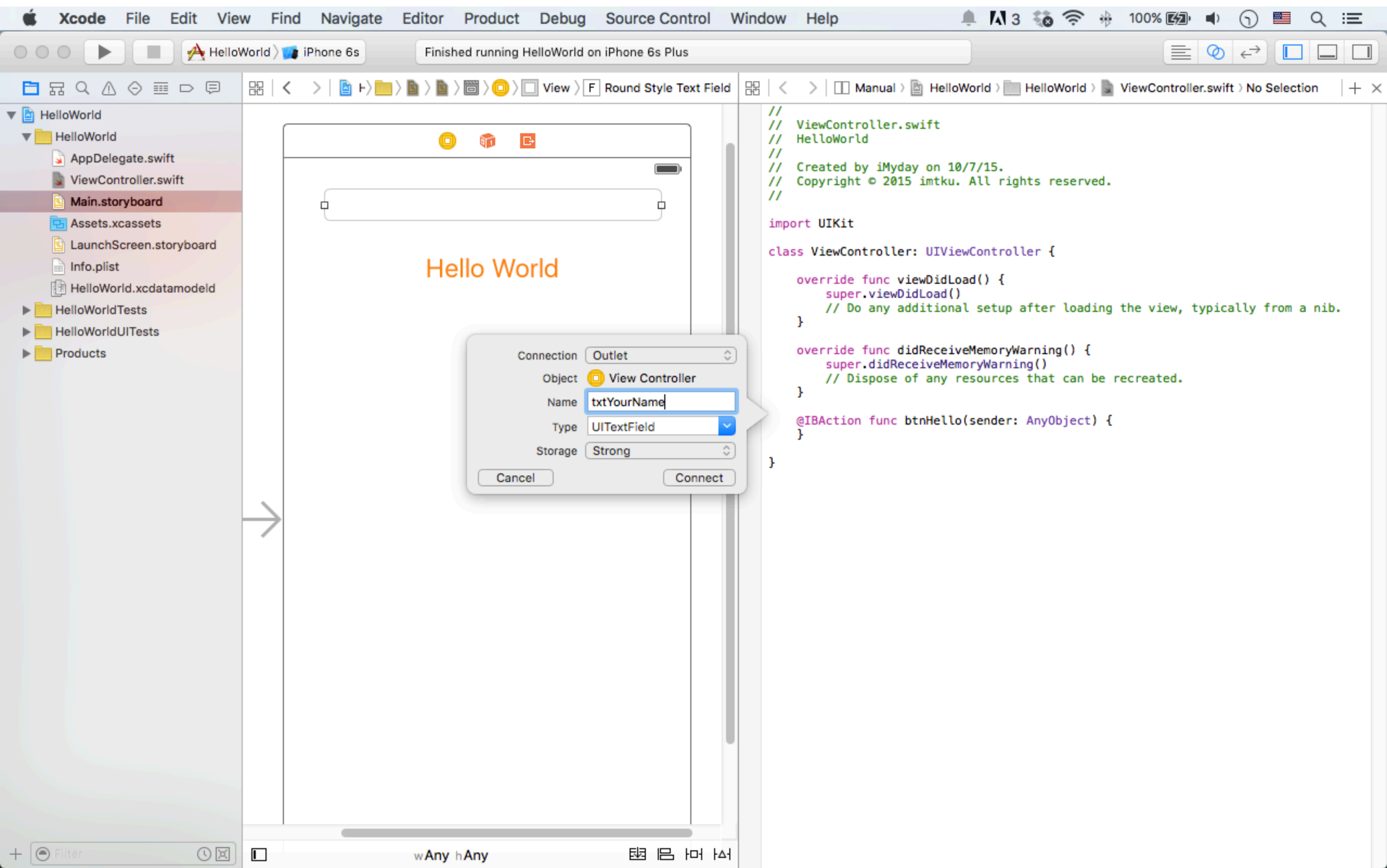
    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view, typically from a nib.
    }

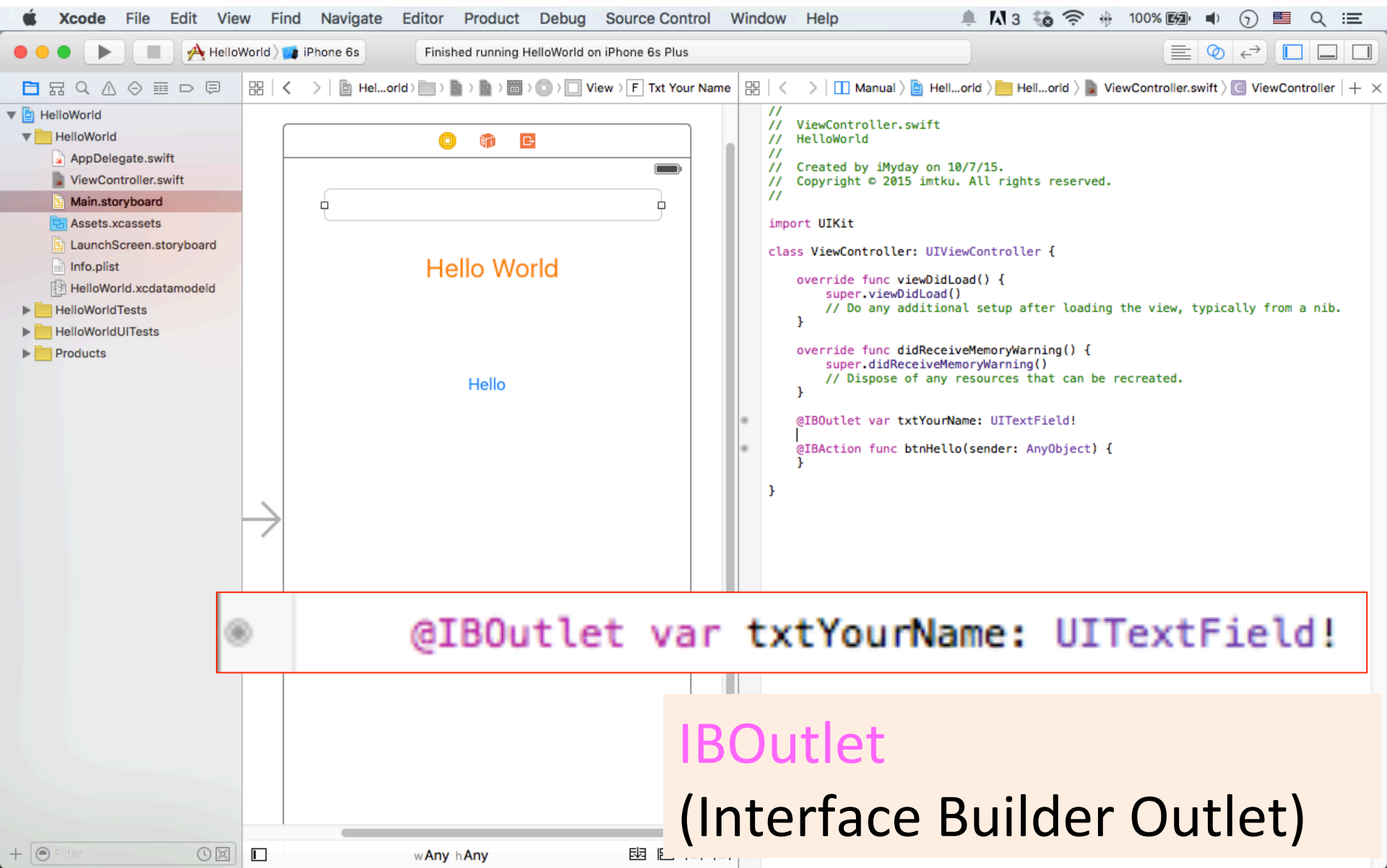
    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }
}
```





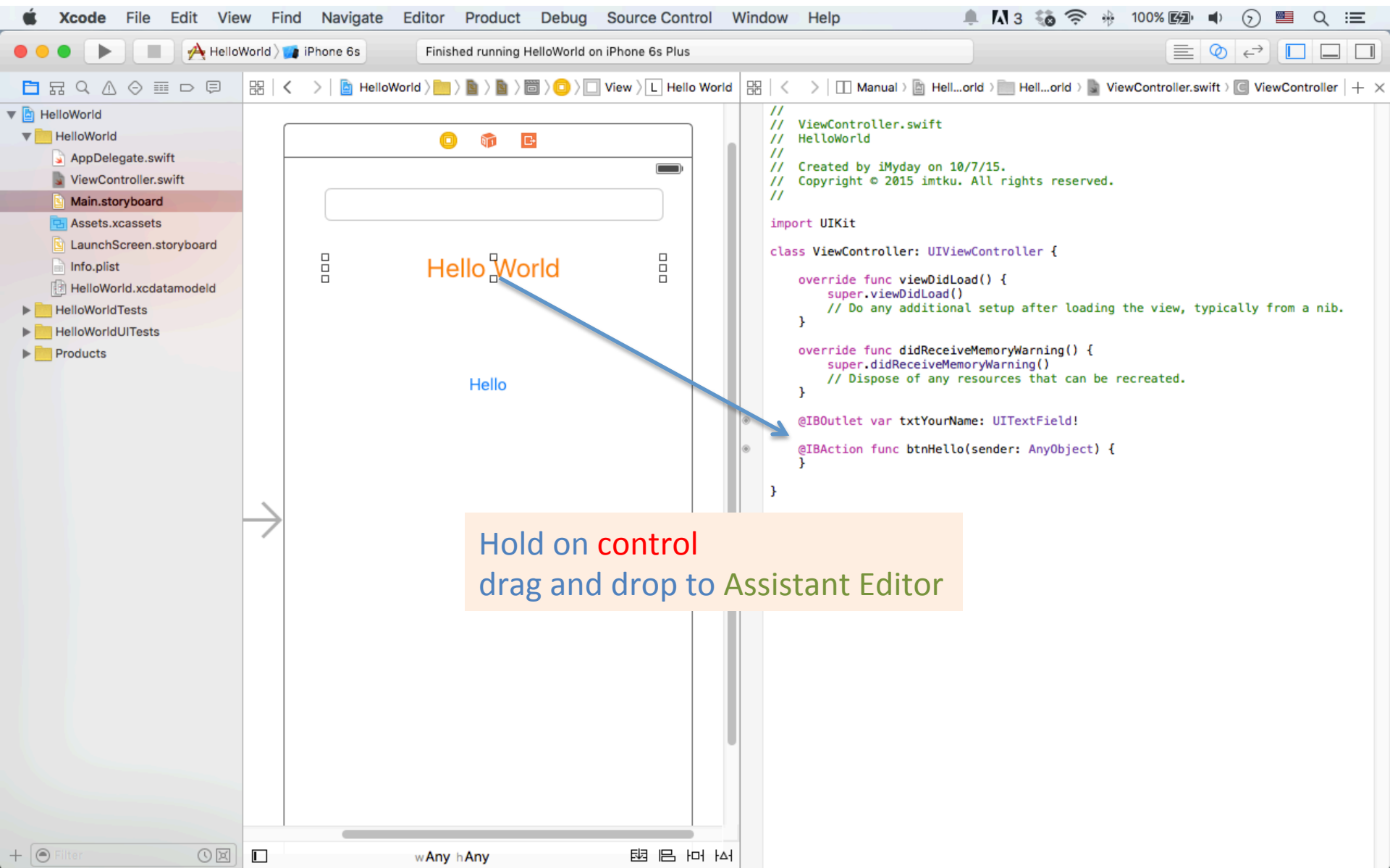


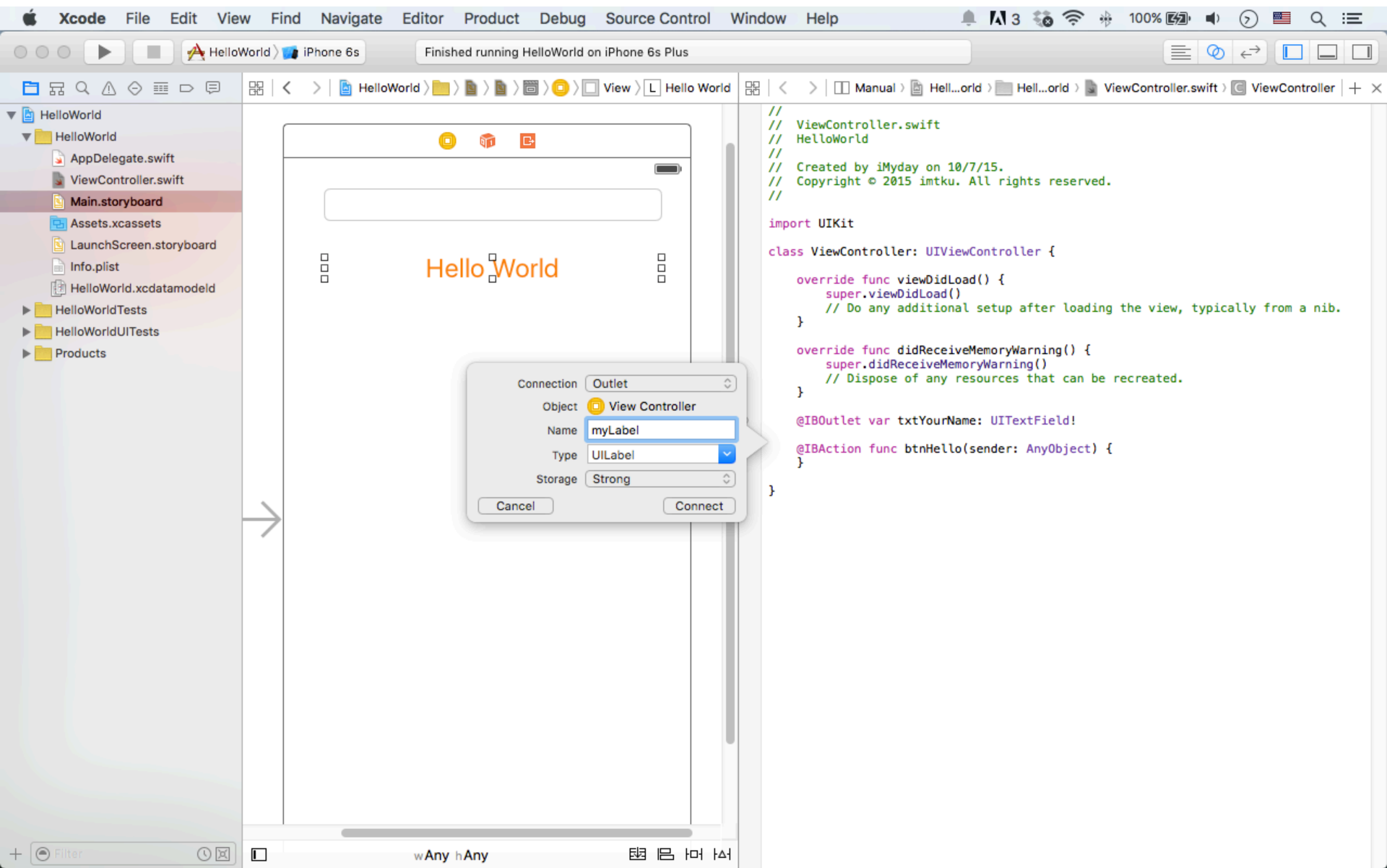


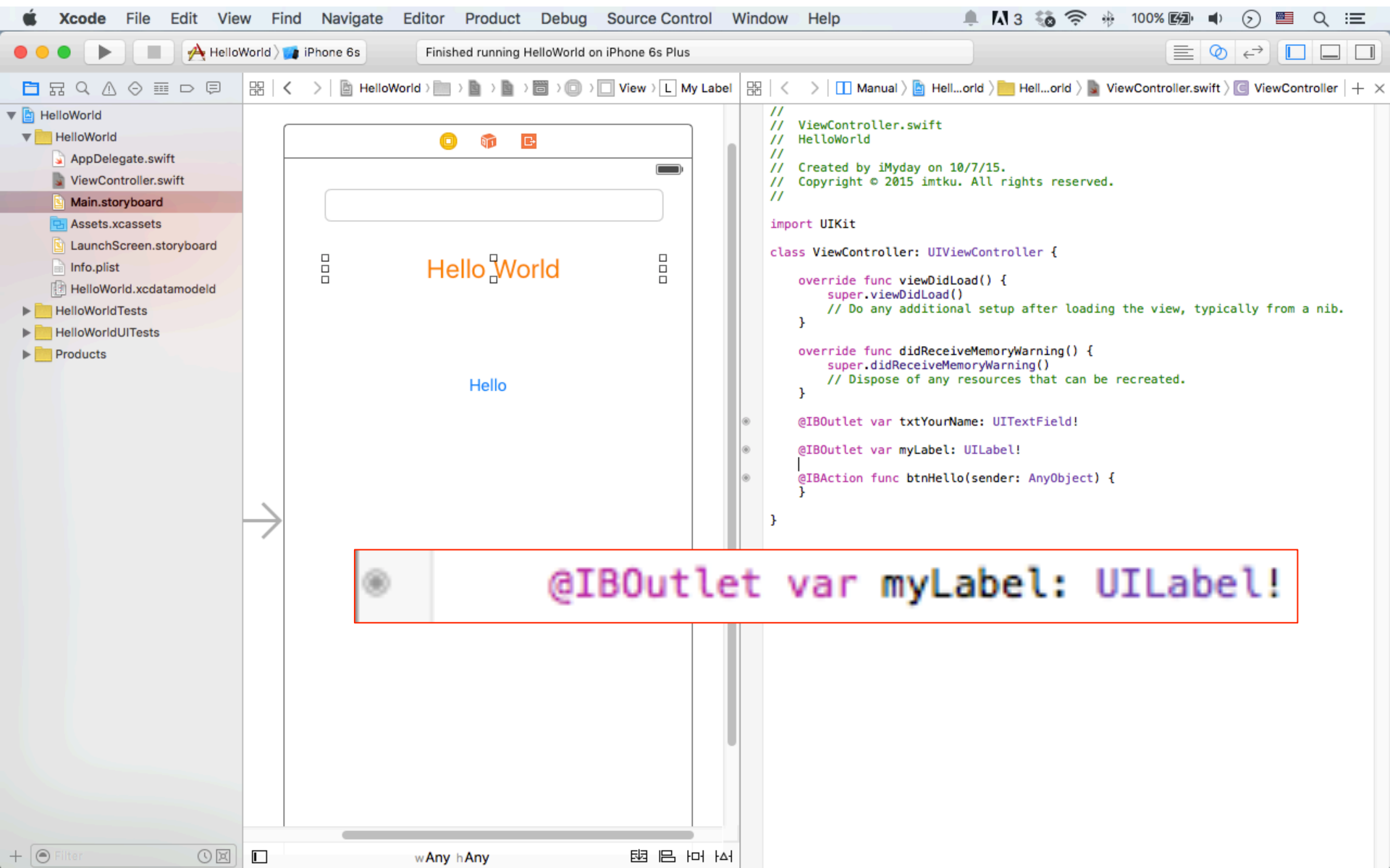


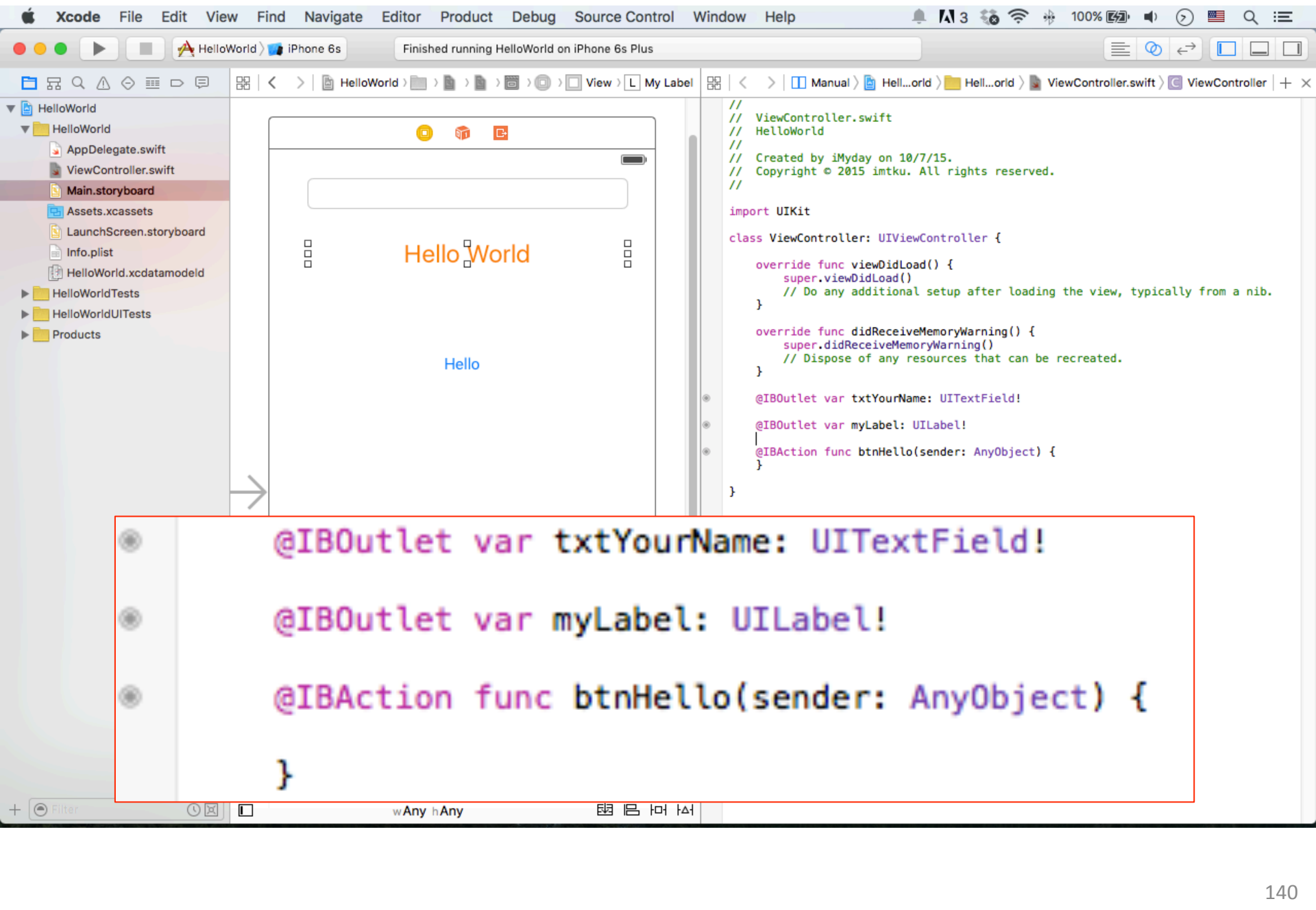
@IBOutlet var txtYourName: UITextField!

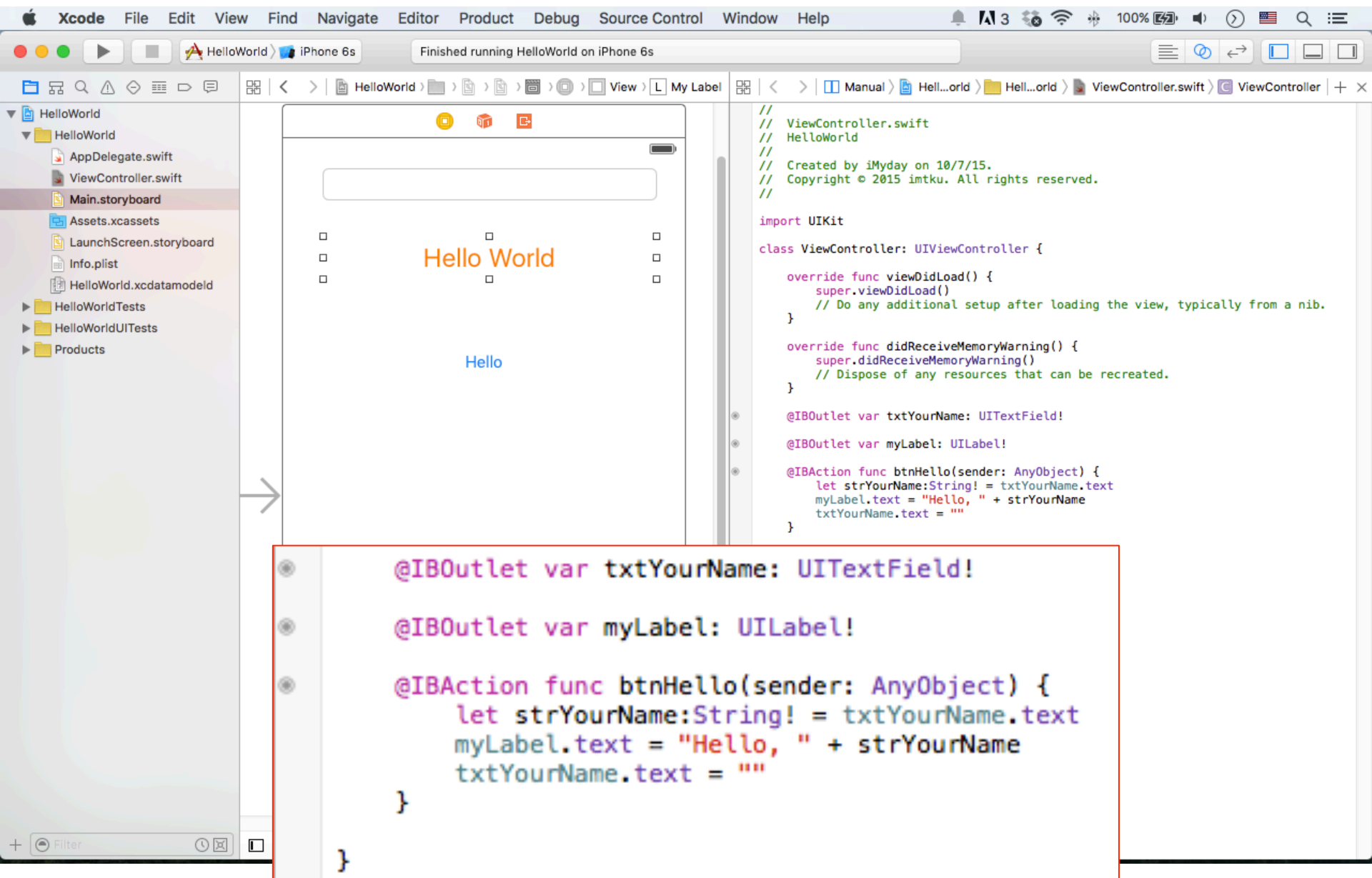
IBOutlet
(Interface Builder Outlet)





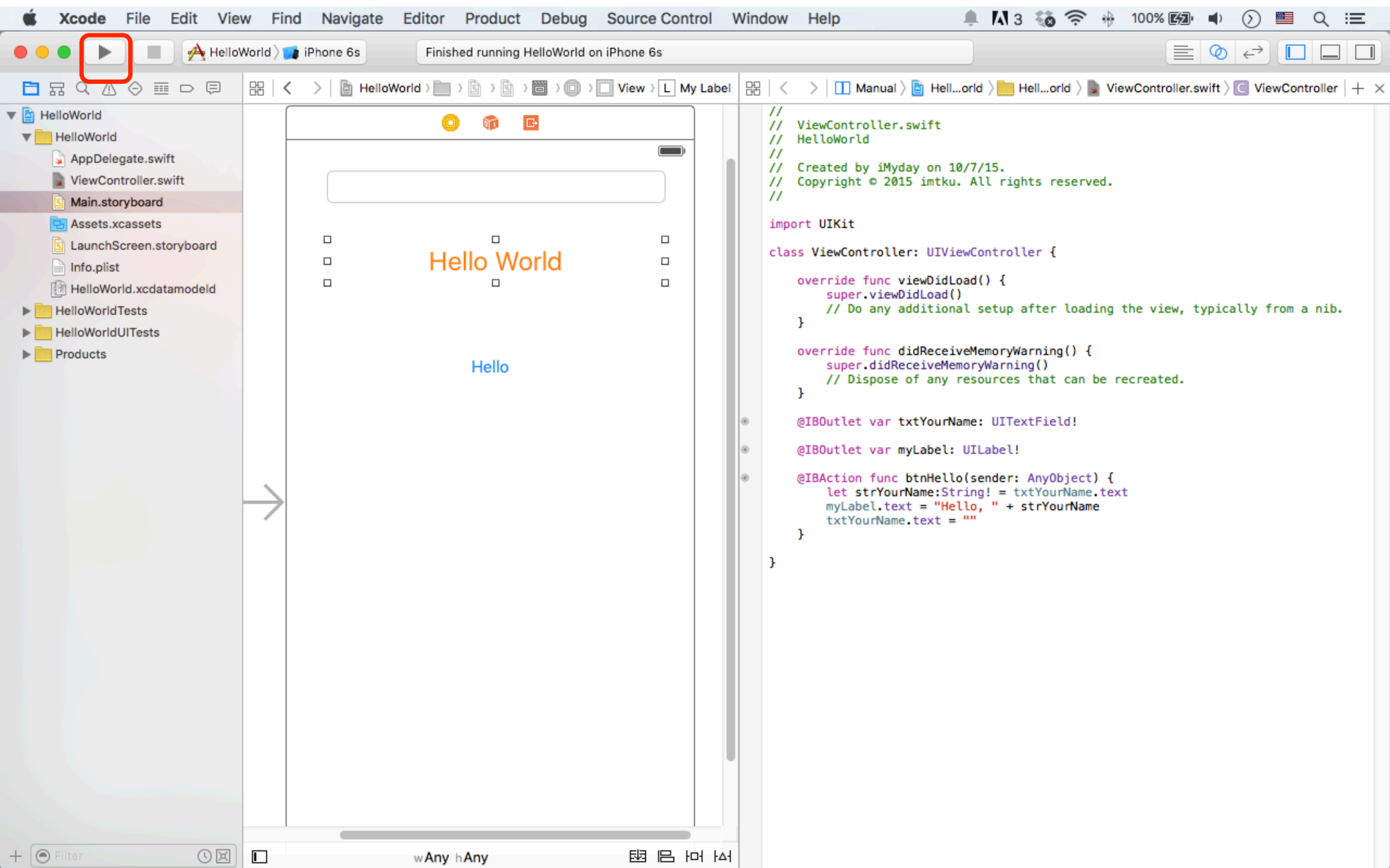






ViewController.swift

```
@IBOutlet var txtYourName: UITextField!  
  
@IBOutlet var myLabel: UILabel!  
  
@IBAction func btnHello(sender: AnyObject) {  
    let strYourName:String! = txtYourName.text  
    myLabel.text = "Hello, " + strYourName  
    txtYourName.text = ""  
}  
  
}
```

Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help

Running HelloWorld on iPhone 6s

View Controller

ViewController.swift

```
// ViewController.swift
// HelloWorld
//
// Created by iMyday on 10/7/15.
// Copyright © 2015 imtku. All rights reserved.

import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view, typically from a nib.
    }

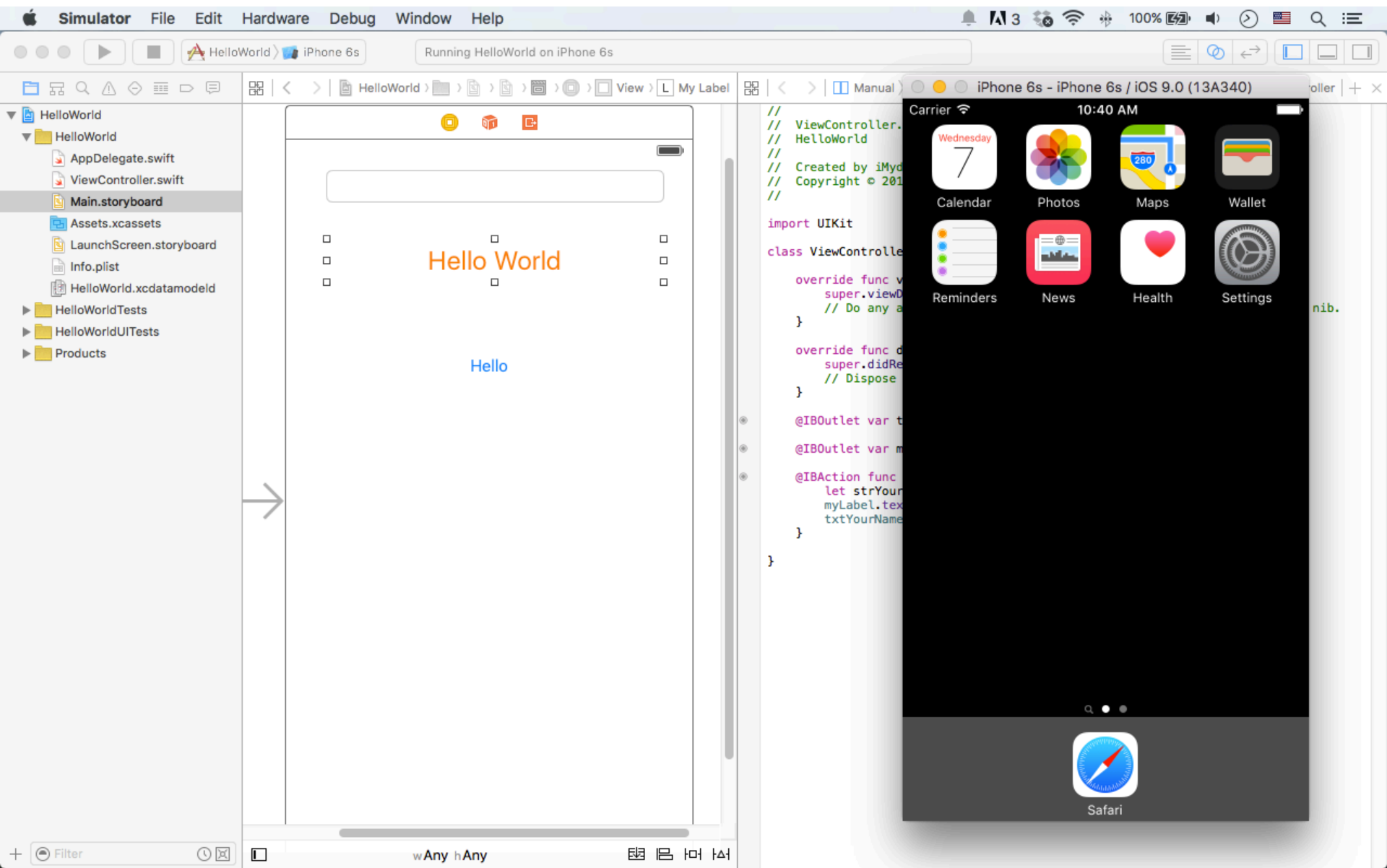
    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }

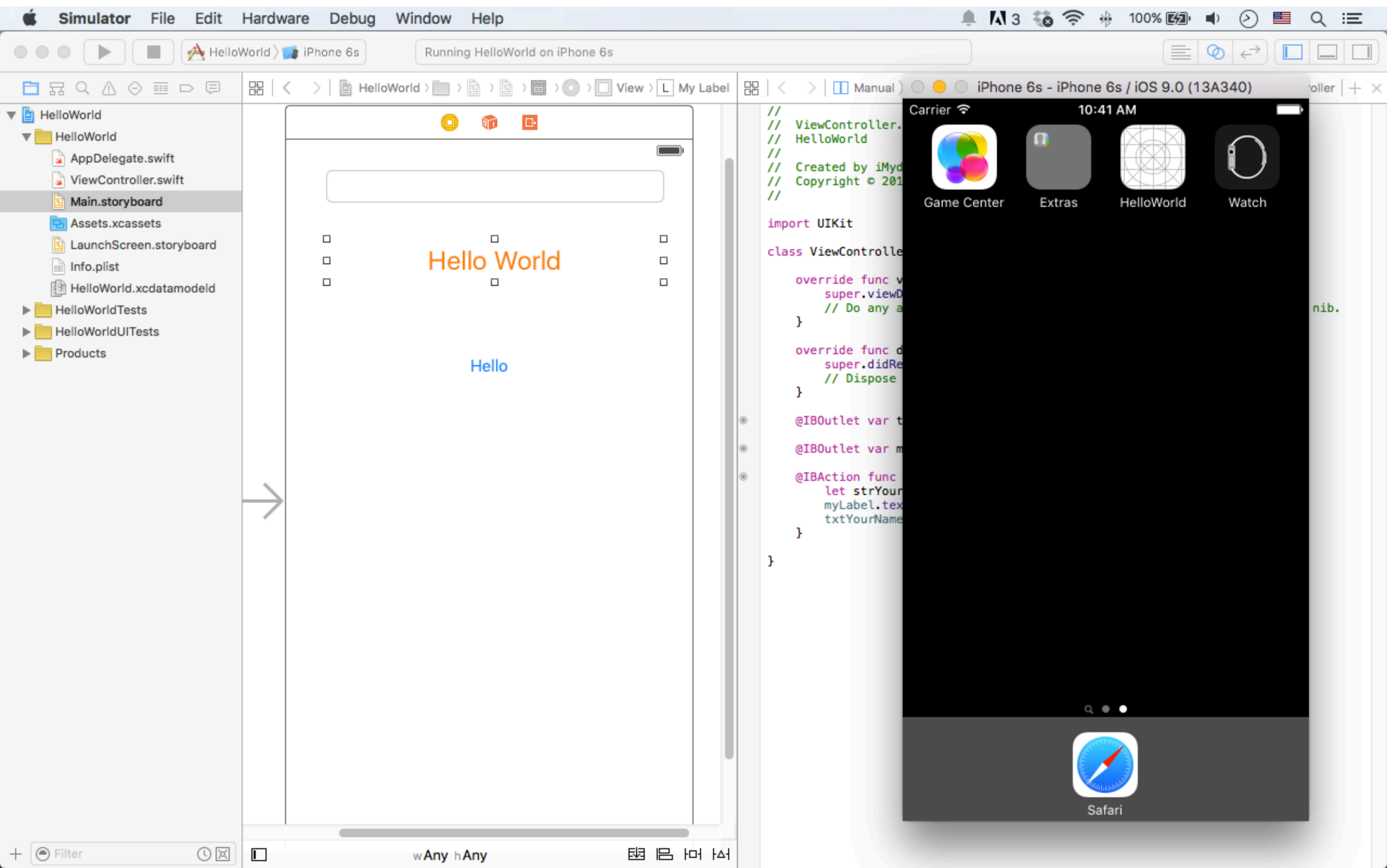
    @IBOutlet var txtYourName: UITextField!

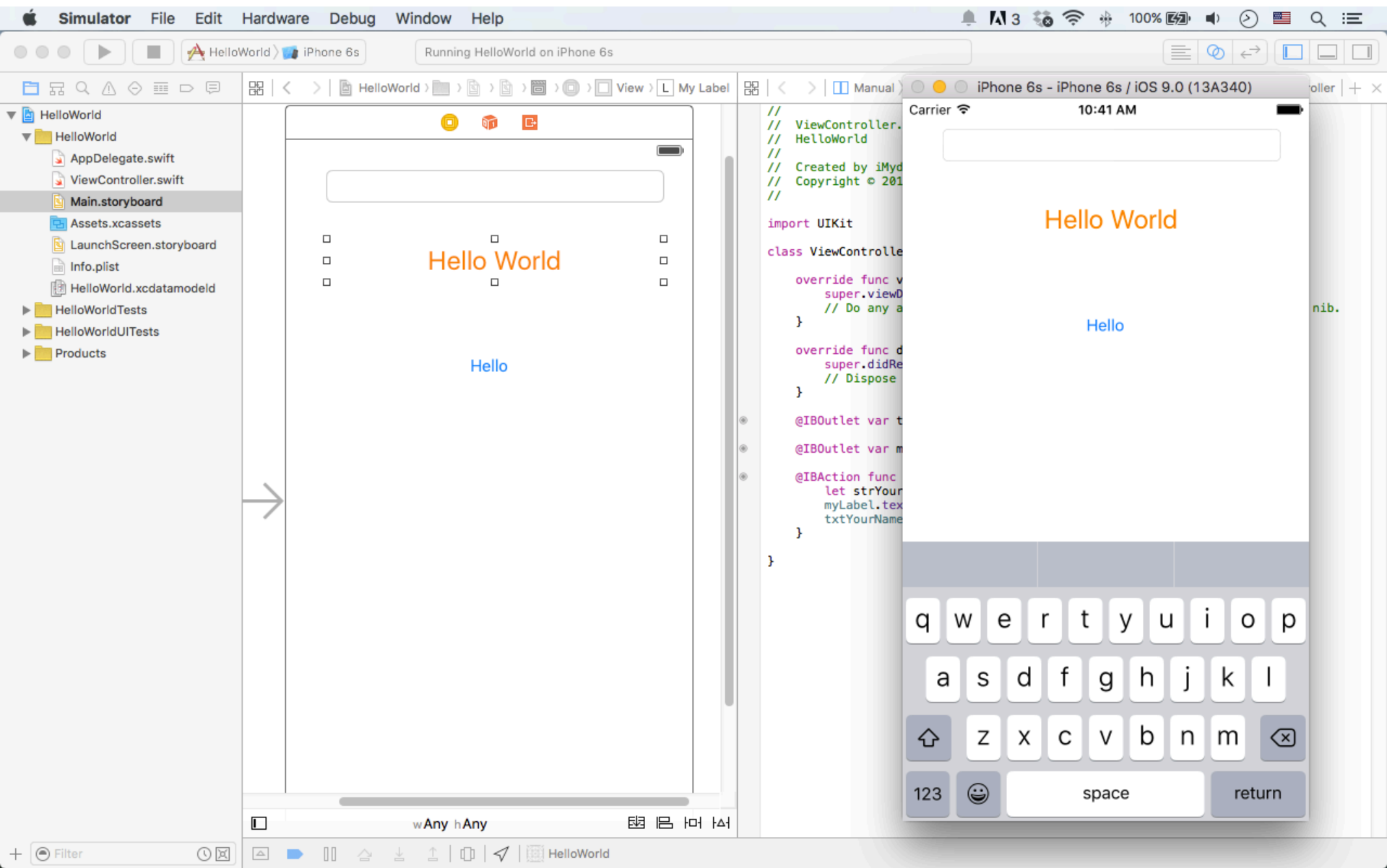
    @IBOutlet var myLabel: UILabel!

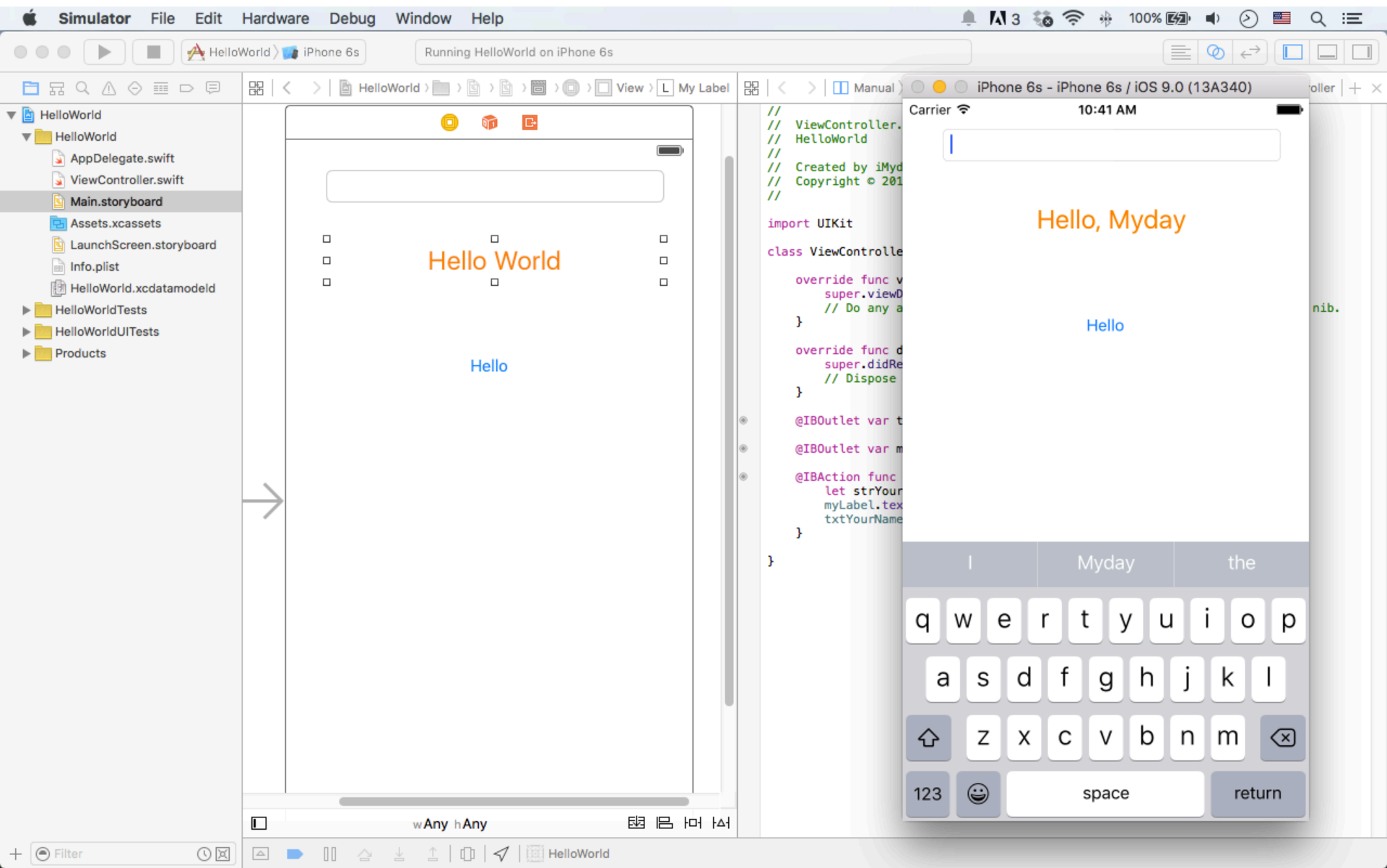
    @IBAction func btnHello(sender: AnyObject) {
        let strYourName:String! = txtYourName.text
        myLabel.text = "Hello, " + strYourName
        txtYourName.text = ""
    }
}
```

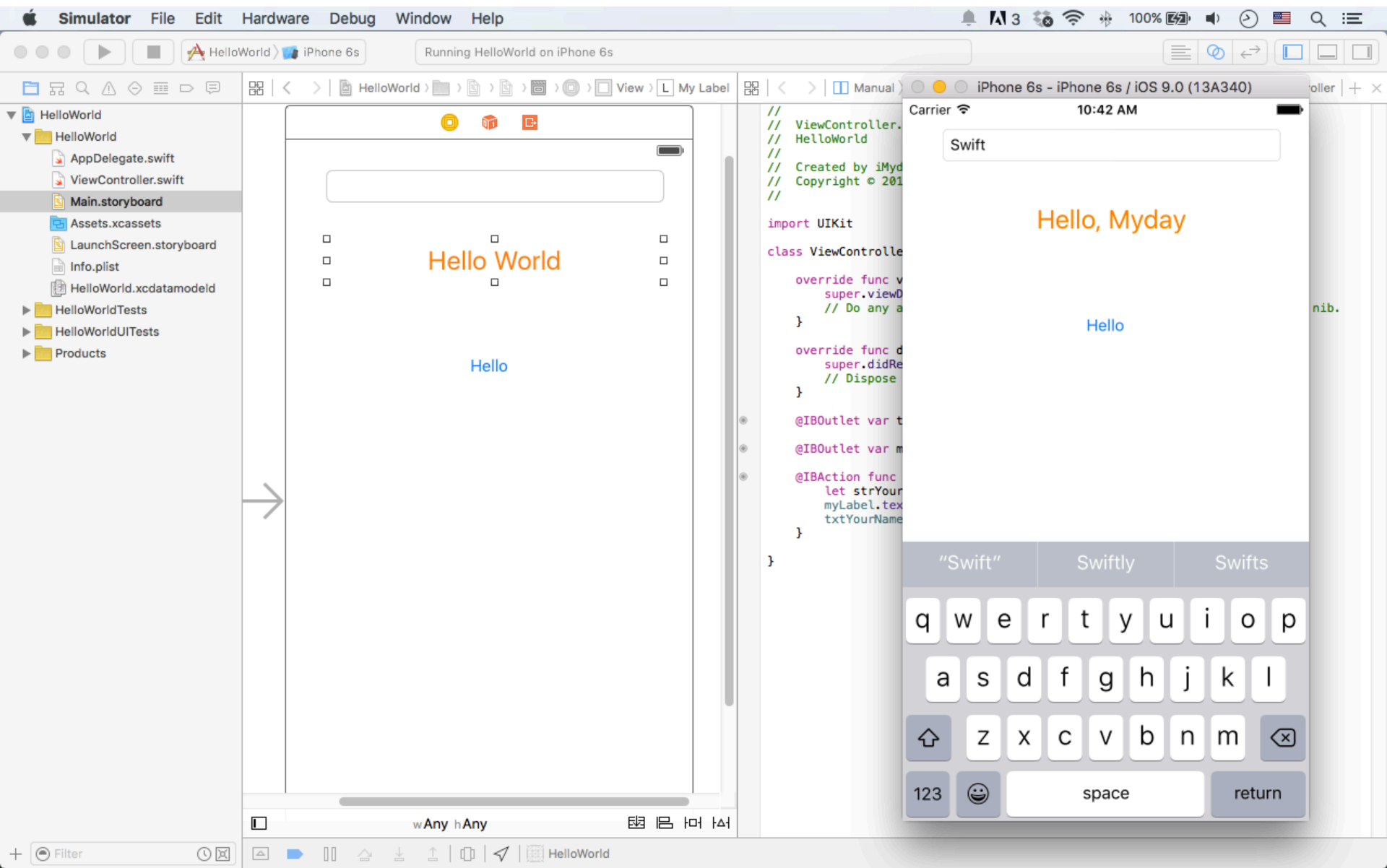
Build Succeeded

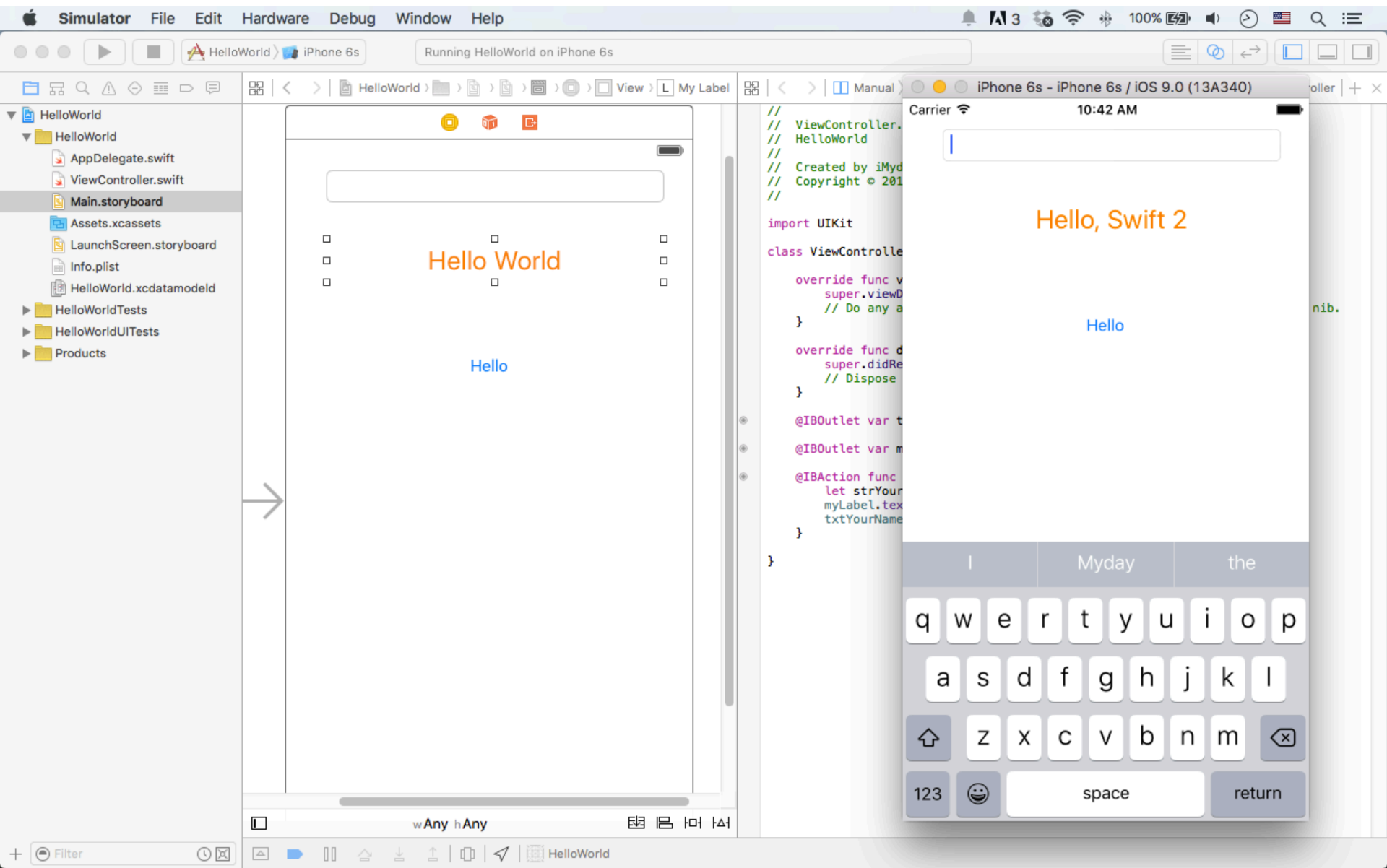


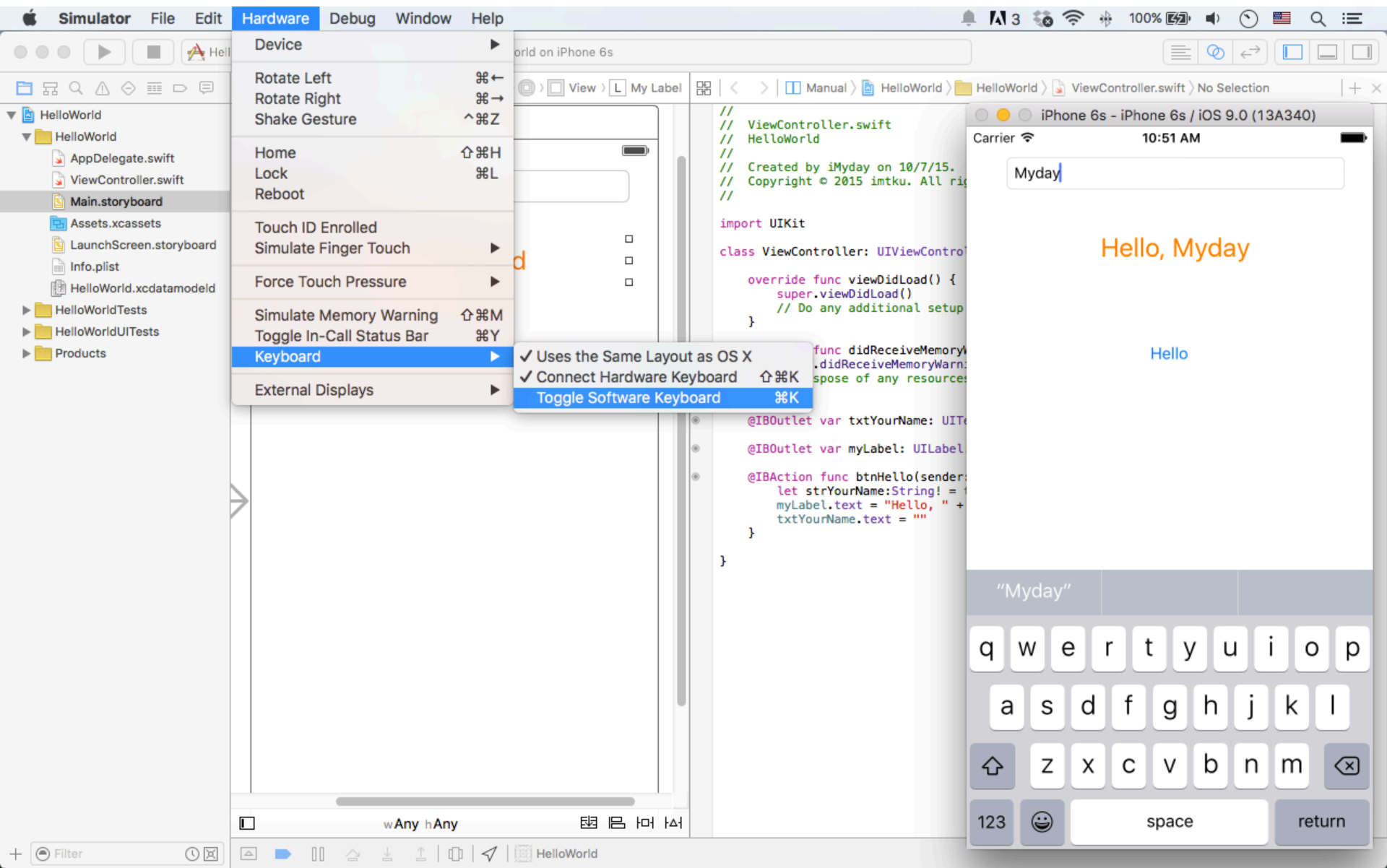


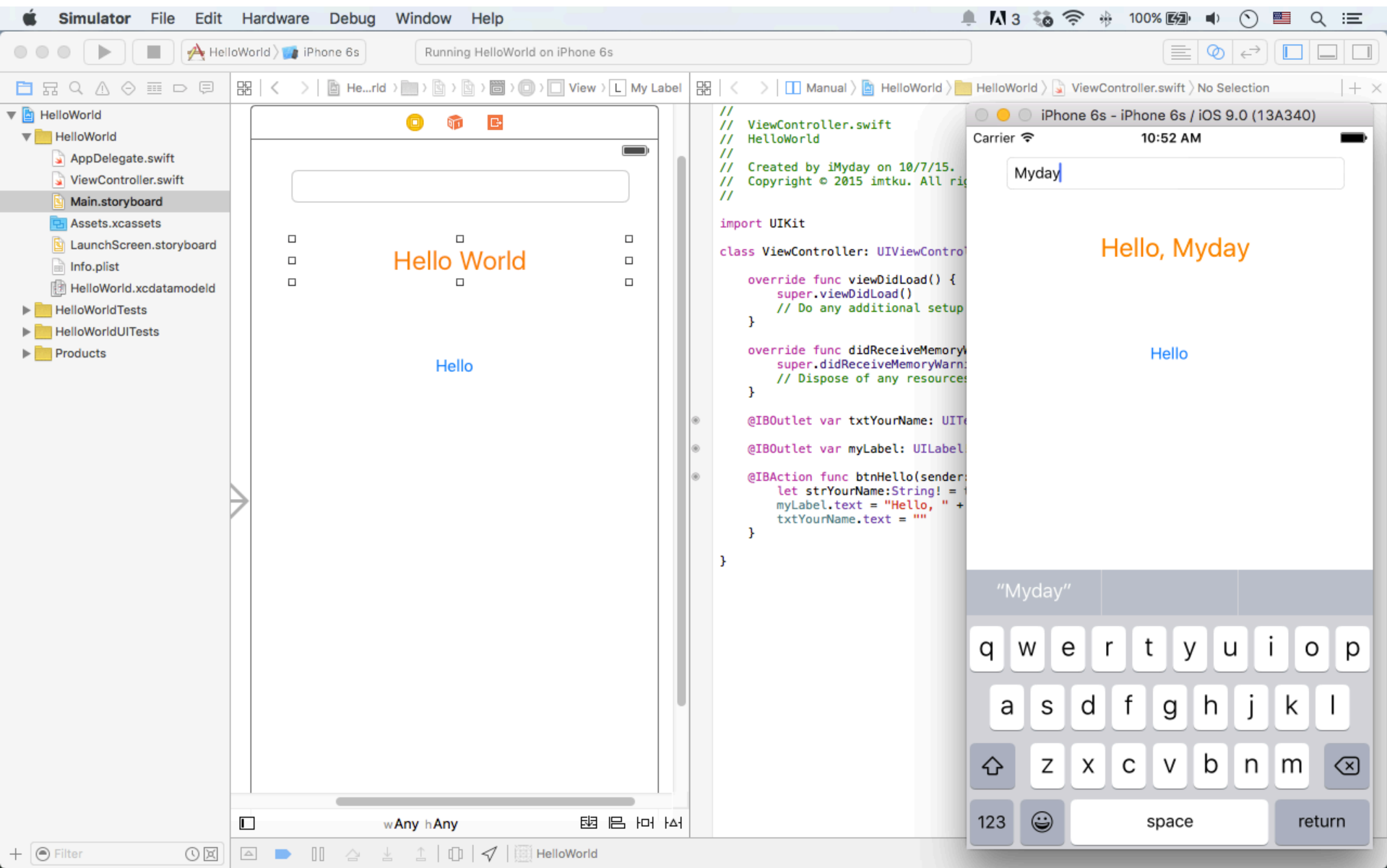




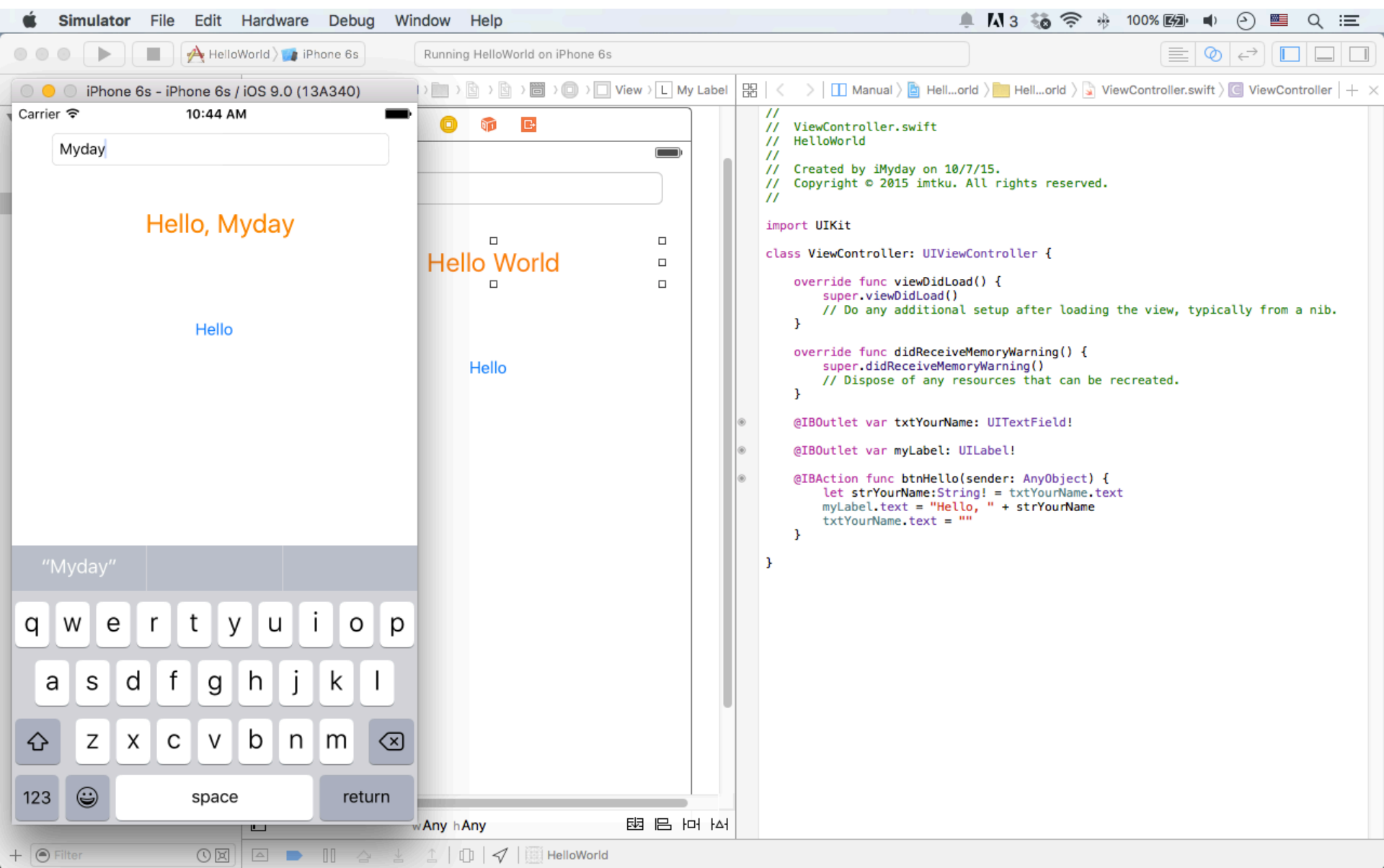








Building Your First iOS App with Xcode 7



Building Your First iOS App with Xcode 7

File Browser (Left):

- HelloWorld (2 targets, iOS SDK 8.0)
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Images.xcassets
 - LaunchScreen.xib
 - HelloWorld.xcdatamodeld
 - Supporting Files
 - HelloWorldTests
 - Products

Storyboard (Center):

- Text Field (Placeholder: Myday)
- Label (Text: Hello World)
- Label (Text: Hello)

Code Editor (Right):

```
// ViewController.swift
// HelloWorld
// Created by iMyday on 2014/10/8.
// Copyright (c) 2014年 imtku. All rights reserved.

import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be
    }

    @IBOutlet var txtYourName: UITextField!
    @IBOutlet var myLabel: UILabel!

    @IBAction func btnHello(sender: AnyObject) {
        myLabel.text = "Hello, " + txtYourName.
    }
}
```

Simulator (Inset):

- Carrier: Wi-Fi
- Time: 8:50 AM
- Text Field: Myday
- Label: Hello, Myday
- Label: Hello
- Keyboard: "Myday", QWERTYUIOP, ASDFGHJKL, ZXCVBNM, 123, space, return

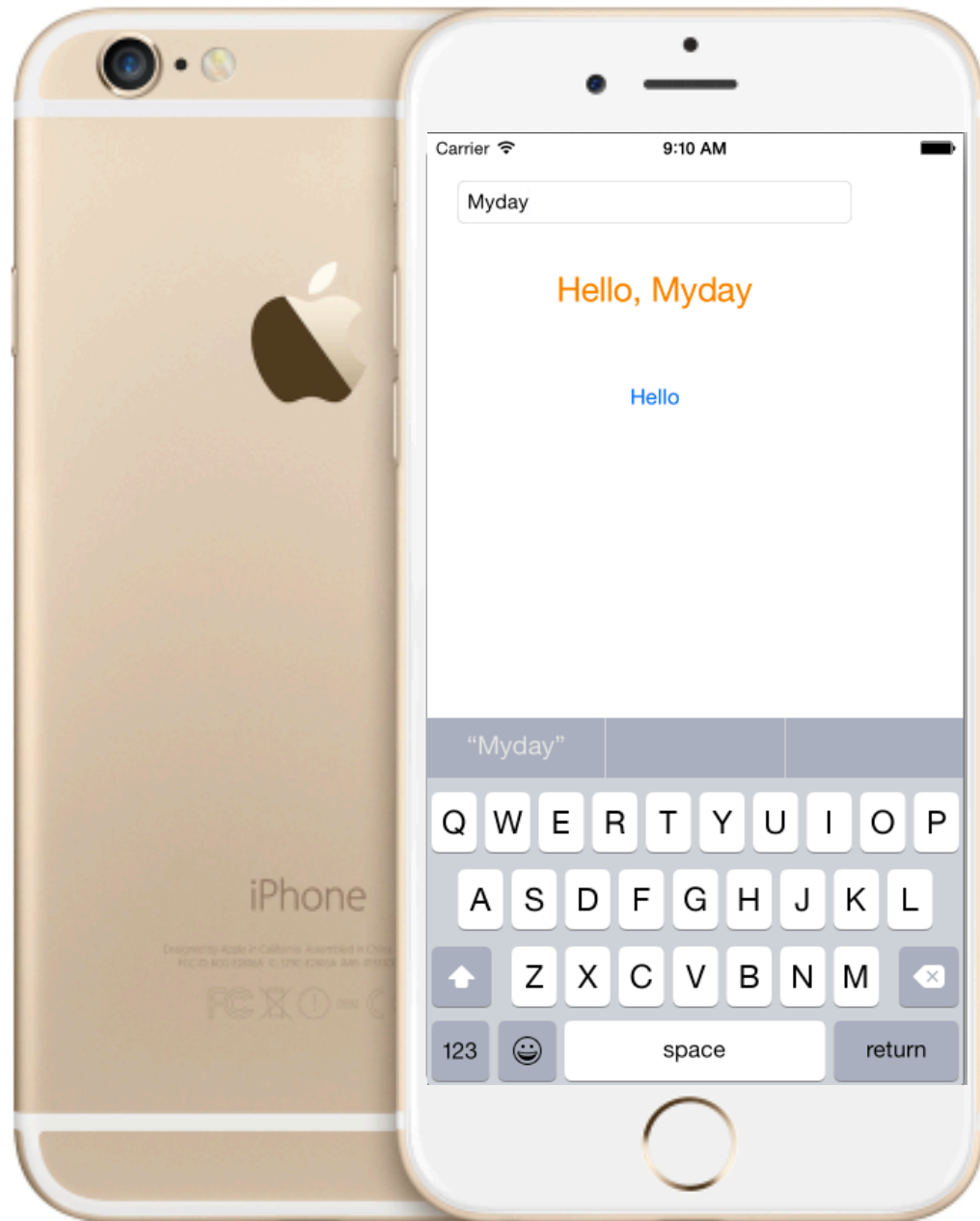
ViewController.swift

```
//  
// ViewController.swift  
// HelloWorld  
//  
// Created by iMyday on 10/7/15.  
// Copyright © 2015 imtku. All rights reserved.  
//  
  
import UIKit  
  
class ViewController: UIViewController {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any additional setup after loading the view, typically from a nib.  
    }  
  
    override func didReceiveMemoryWarning() {  
        super.didReceiveMemoryWarning()  
        // Dispose of any resources that can be recreated.  
    }  
  
    @IBOutlet var txtYourName: UITextField!  
    @IBOutlet var myLabel: UILabel!  
    @IBAction func btnHello(sender: AnyObject) {  
        let strYourName:String! = txtYourName.text  
        myLabel.text = "Hello, " + strYourName  
        txtYourName.text = ""  
    }  
  
}
```

ViewController.swift

```
@IBOutlet var txtYourName: UITextField!  
@IBOutlet var myLabel: UILabel!  
@IBAction func btnHello(sender: AnyObject) {  
    let strYourName:String! = txtYourName.text  
    myLabel.text = "Hello, " + strYourName  
    txtYourName.text = ""  
}
```





Summary

- **Developing iPhone / iPad Native Apps with Swift 2 (Xcode 7)**
 - Mac OS X 10.8, 10.9, 10.10, 10.11
 - Xcode 6, Xcode 7
 - iOS 8, iOS 9
- **Building Your First iOS App with Xcode 7**

References

- Apple Developers, <https://developer.apple.com/>
- Start Developing iOS Apps Today,
 - <https://developer.apple.com/library/ios/referencelibrary/GettingStarted/RoadMapiOS/>
 - <https://developer.apple.com/library/ios/referencelibrary/GettingStarted/RoadMapiOS/RoadMapiOS.pdf>
- Rory Lewis and Laurence Moroney, **iPhone and iPad Apps for Absolute Beginners**, 4th Edition, Apress, 2013
- The Swift Programming Language (Swift 2),
 - <https://itunes.apple.com/us/book/the-swift-programming-language/id881256329>
 - https://developer.apple.com/library/ios/documentation/Swift/Conceptual/Swift_Programming_Language/
- Apple - WWDC 2014
 - <https://www.youtube.com/watch?v=w87fOAG8fjk>
- Apple - WWDC 2015
 - https://www.youtube.com/watch?v=_p8AsQhaVKI