Social Media Apps Programming

Developing iPhone / iPad Native Apps with Swift (XCode)

Min-Yuh Day, Ph.D.
Assistant Professor
Department of Information Management
Tamkang University

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

2015-10-07
<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Subject/Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2015/09/16</td>
<td>Course Orientation and Introduction to Social Media and Mobile Apps Programming</td>
</tr>
<tr>
<td>2</td>
<td>2015/09/23</td>
<td>Introduction to Android / iOS Apps Programming</td>
</tr>
<tr>
<td>3</td>
<td>2015/09/30</td>
<td>Developing Android Native Apps with Java (Android Studio) (MIT App Inventor)</td>
</tr>
<tr>
<td>4</td>
<td>2015/10/07</td>
<td>Developing iPhone / iPad Native Apps with Swift (XCode)</td>
</tr>
<tr>
<td>5</td>
<td>2015/10/14</td>
<td>Mobile Apps using HTML5/CSS3/JavaScript</td>
</tr>
<tr>
<td>6</td>
<td>2015/10/21</td>
<td>jQuery Mobile</td>
</tr>
<tr>
<td>Week</td>
<td>Date</td>
<td>Subject/Topics</td>
</tr>
<tr>
<td>------</td>
<td>------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>7</td>
<td>2015/10/28</td>
<td>Create Hybrid Apps with Phonegap</td>
</tr>
<tr>
<td>8</td>
<td>2015/11/04</td>
<td>jQuery Mobile/Phonegap</td>
</tr>
<tr>
<td>9</td>
<td>2015/11/11</td>
<td>jQuery Mobile/Phonegap</td>
</tr>
<tr>
<td>10</td>
<td>2015/11/18</td>
<td>Midterm Exam Week (Midterm Project Report)</td>
</tr>
<tr>
<td>11</td>
<td>2015/11/25</td>
<td>Case Study on Social Media Apps Programming and Marketing in Google Play and App Store</td>
</tr>
<tr>
<td>12</td>
<td>2015/12/02</td>
<td>Google Cloud Platform</td>
</tr>
<tr>
<td>Week</td>
<td>Date</td>
<td>Subject/Topics</td>
</tr>
<tr>
<td>------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>13</td>
<td>2015/12/09</td>
<td>Google App Engine</td>
</tr>
<tr>
<td>14</td>
<td>2015/12/16</td>
<td>Google Map API</td>
</tr>
<tr>
<td>15</td>
<td>2015/12/23</td>
<td>Facebook API (Facebook JavaScript SDK) (Integrate Facebook with iOS/Android Apps)</td>
</tr>
<tr>
<td>16</td>
<td>2015/12/30</td>
<td>Twitter API</td>
</tr>
<tr>
<td>17</td>
<td>2016/01/06</td>
<td>Final Project Presentation</td>
</tr>
<tr>
<td>18</td>
<td>2016/01/13</td>
<td>Final Exam Week (Final Project Presentation)</td>
</tr>
</tbody>
</table>
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
Building Your First iOS App with Xcode 7
Android /iOS Apps Programming

Native Apps

Hybrid Apps

Mobile Web Apps
App Development Comparison

Native Apps
- Full Access
- Very Fast Speed
- Expensive Development Cost
- Available App Store
- Mandatory Approval Process

Hybrid Apps
- Full Access
- Native Speed as Necessary Speed
- Reasonable Development Cost
- Available App Store
- Low Overhead Approval Process

Web Apps
- Partial Access
- Fast Speed
- Reasonable Development Cost
- Not Available App Store
- None Approval Process

iOS - Native App Development

Native App – Interaction with Mobile Device

Apple App Distribution Workflows

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.

Source: https://developer.apple.com/
Download the latest development tools and SDKs.

iOS Apps
Mac Apps
Xcode

Source: https://developer.apple.com/
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 ›

Source: https://developer.apple.com/

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.

400 Pages
User Level: Beginner
Publishing October 23, 2013, but available now as part of the *Alpha Program*

http://www.apress.com/9781430263616
Start Developing iOS Apps Today

Setup

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 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.

Start Developing iOS Apps Today

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**

The notebook people love.

MacBook Air

Source: [https://www.apple.com/mac/](https://www.apple.com/mac/)
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 notebook⁶

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 life
- Up to 1TB 5400–rpm hard drive; or up to 512GB solid–state drive
- 4.5 pounds

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 life
- Up to 1TB flash storage
- 3.48 pounds

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 life
- Up to 1TB flash storage
- 4.49 pounds

Source: http://www.apple.com/mac/compare/
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)
- 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)
- 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)
- Apple Wireless Keyboard and Magic Mouse or Magic Trackpad

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

Source: http://www.apple.com/mac/compare/
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.

Source: http://www.apple.com/osx/
OS X El Capitan
A refined experience and improved performance for your Mac.

Best New Apps & Games

OS X El Capitan
Utilities
⭐⭐⭐⭐⭐
165 Ratings
GET
OS X El Capitan
A refined experience and improved performance for your Mac.

Learn More ›
OS X El Capitan for Developers

https://developer.apple.com/osx/
Xcode 6

The complete toolset for building great apps.

Source: https://developer.apple.com/xcode/
**Xcode 7**

The complete toolset for building great apps.

OS X Yosemite for Developers

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

Source: https://developer.apple.com/ios8/
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.

Source: https://developer.apple.com/ios/
Xcode 6

Source: https://developer.apple.com/xcode/
Get the Tools

Mac App Store → Xcode (FREE) → Download

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

Source: https://developer.apple.com/xcode/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

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 >
Xcode

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.

What's New in Version 7.0.1
Xcode 7.0.1 fixes bugs related to app thinning.

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.r.l.
© 1996-2014 Apple Inc.
Rated 4+
Compatibility: OS X 10.10.x or later

More Apps by Apple
Xcode
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)

http://www.appcoda.com/learnsswift/get-started.html
Swift.
A modern programming language that is safe, fast, and interactive.

https://developer.apple.com/swift/
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.

Source: [https://developer.apple.com/swift/](https://developer.apple.com/swift/)
Swift Language

• Swift is a new object-oriented programming language for iOS and OS X development.

• Swift is modern, powerful, expressive, and easy to use.

Source: https://developer.apple.com/swift/
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

Source: https://developer.apple.com/swift/
print("Hello world")
println("Hello world")

println("Hello World")

'println' has been renamed to 'print'

Fix-it Replace "print" with "print"

print("Hello world")
let count = 10
var price = 23.55

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

print(message)
const int count = 10;
double price = 23.55;

NSString *firstMessage = @"Swift is awesome. ";
NSString *secondMessage = @"What do you think?";
NSString *message = [NSString stringWithFormat:@"%@%@", 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)
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: MyPlayground
Platform: iOS
Swift 2 in Xcode 7 Playground

```swift
//: Playground - noun: a place where people can play

import UIKit

var str = "Hello, playground"
```
Swift 2 in Xcode 7 Playground

```
//: 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")
}
```

"Hello, playground"
"Hello World"
"Hello World\n"
7
2
9
"Hello Swift 2\n"
`var message = "Hello World"
print(message)

var a = 7
var b = 2
var total = a + b

if (total < 10) {
    print("Hello Swift 2")
}
iOS App Development Process

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

iOS App Development Process

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

Objects Communicate Through Messages

Current Execution Point

... [somePerson sayHello];
...

XYZPerson Implementation

@implementation XYZPerson
-(void)sayHello {
    NSLog(@"Hello, world!");
}
@end

Protocols Define Messaging Contracts

Designing a User Interface

Model → Controller → View

Use Storyboards to Lay Out Views

Defining the Interaction

Model

Controller

View

View Controllers

Application controller layer

View layer

Window → View → Image View, Text View, Button

Use Storyboards to Define Navigation

Segue

Storyboards

Incorporating the Data

Using Design Patterns

Model-View-Controller (MVC)

Target-Action

UIControlEventTouchUpInside

Restore Defaults

aControl

action=restoreDefaults:
target=controller

Controller

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
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
Choose a template for your new project:

iOS
Application
Framework & Library
watchOS
Application
Framework & Library
OS X
Application
Framework & Library
System Plug-In
Other

Master-Detail Application
Page-Based Application
Single View Application
Tabbed Application

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.
Swift Language
Main.storyboard (UI)
ViewController.swift (Code)
Main.storyboard (UI)
Text Field
Label

Label - A variable-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 tapped.
Hello World
Button

Hello World
Hello World
Build and Run
Hello World
Hello World
Hello World

Hello
Main.storyboard (UI)
ViewController.swift (Code)
ViewController.swift (Code)
IBOutlet and IBAction

• IBOutlet
  – Interface Builder Outlet

• IBAction
  – Interface Builder Action
Hello World
Hello World

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.
    }

}`
Hello World

```swift
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.
}
```

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.
    }
}
Hold on control drag and drop button to Assistant Editor
Hello World

```swift
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.
    }

    // ViewController.swift
    // HelloWorld
    //
    // Created by JMyday on 10/7/15.
    // Copyright © 2015 JMyday. 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.
        }
    }
```
Hello World

// ViewController.swift
// HelloWorld
//
// Created by iMyday on 10/7/15.
// Copyright © 2015 intku. 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.
    }
}
Hello World

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.
}
@IBAction func btnHello(sender: AnyObject) {
}

IBActionAction
(Interface Builder Action)
Hold on **control** drag and drop to Assistant Editor
Hello World

```
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.
    }

    @IBAction func btnHello(sender: AnyObject) {
    }
```

IBOutlet (Interface Builder Outlet)
Hold on control drag and drop to Assistant Editor
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 txtName: UITextField!
    @IBAction func btnHello(sender: AnyObject) {
    }
}
@IBOutlet var myLabel: UILabel!
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) {
    }
}
// ViewController.swift
// HelloWorld

// Created by jMyday on 10/7/15.
// Copyright © 2015 jMyday. 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 = ""
    }
}
@IBOutlet var txtYourName: UITextField!

@IBOutlet var myLabel: UILabel!

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

Hello
Hello World

Hello
Hello, Myday
Hello World

Hello, Myday

Myday
Building Your First iOS App with Xcode 7
Building Your First iOS App with Xcode 7
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 = ""
    }
}
@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

• 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

  – The Swift Programming Language (Swift 2),

  – Apple - WWDC 2014
    • https://www.youtube.com/watch?v=w87fOAG8fjk

  – Apple - WWDC 2015
    • https://www.youtube.com/watch?v=_p8AsQhaVKI