



# iOS 앱 개발 환경





# 학습목표

- ios앱 개발도구인 xcode를 이용하여 프로젝트를 생성할 수 있다.
- MVC가 무엇인지 이해하고, xcode와 연관관계를 이해한다.
- Xcode를 이용하여 간단한 UI를 작성할 수 있다.
- Xcode에서 M, V, C 관계를 이해하고, 상호 연결 방법을 이해한다



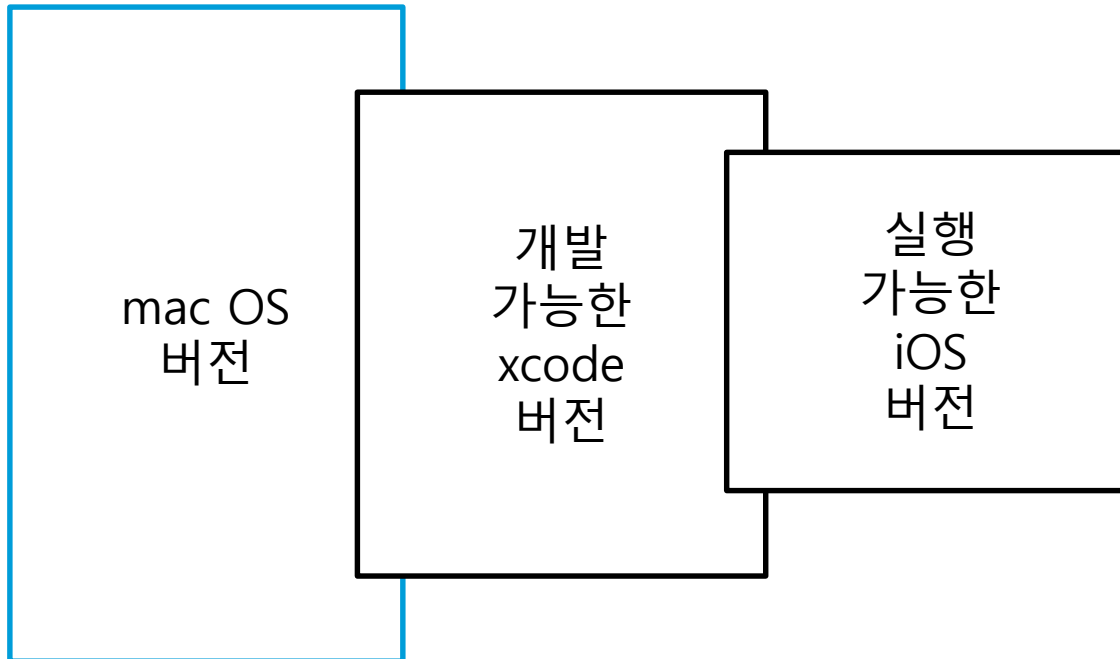
# 개발 환경

## ■ 맥컴퓨터

- 반드시 맥에서만 개발할 수 있음

## ■ xcode

- 대부분 xcdoe를 사용하여야 개발할 수 있음
- 안드로이드 앱 개발을 위하여 안드로이드 스튜디오를 사용하듯 iOS 앱을 개발하기 위해서는 반드시 xcode를 사용하여야 함





## ■ macOS별 개발환경


- <https://developer.apple.com/support/xcode/>

## Minimum requirements and supported SDKs

Xcode Version	Minimum OS Required	SDK	Architecture	Deployment Targets		Simulator	Swift			
Xcode 15.1 beta**	macOS Ventura 13.5	iOS 17 macOS 14 tvOS 17 watchOS 10 DriverKit 23 visionOS 1	i386 x86_64 armv7k arm64 arm64e arm64_32	Xcode 13.2	macOS Big Sur 11.3	iOS 15.2 macOS 12.1 tvOS 15.2 watchOS 8.3 DriverKit 21.2	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15.2 iPadOS 13-15.2 macOS 10.9-12.2 tvOS 9-15.2 watchOS 2-8.3 DriverKit 19-21.2	iOS 10.3.1-15.2 tvOS 10.2-15.2 watchOS 3.2-8.3	Swift 4 Swift 4.2 Swift 5.5
				Xcode 13.1	macOS Big Sur 11.3	iOS 15 macOS 12 tvOS 15 watchOS 8 DriverKit 210.1	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15 iPadOS 13-15 macOS 10.9-12 tvOS 9-15 watchOS 2-8 DriverKit 19-210.1	iOS 10.3.1-15 tvOS 10.2-15 watchOS 3.2-8	Swift 4 Swift 4.2 Swift 5.5
				Xcode 13	macOS Big Sur 11.3	iOS 15 macOS 11.3 tvOS 15 watchOS 8 DriverKit 20.4	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15 iPadOS 13-15 macOS 10.9-11.3 tvOS 9-15 watchOS 2-8 DriverKit 19-20.4	iOS 10.3.1-15 tvOS 10.2-15 watchOS 3.2-8	Swift 4 Swift 4.2 Swift 5.5
Xcode 15.0.x	macOS Ventura 13.5	iOS 17 macOS 14 tvOS 17 watchOS 10 DriverKit 23	i386 x86_64 armv7k arm64 arm64e arm64_32	Xcode 13.2	macOS Big Sur 11.3	iOS 15.2 macOS 12.1 tvOS 15.2 watchOS 8.3 DriverKit 21.2	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15.2 iPadOS 13-15.2 macOS 10.9-12.2 tvOS 9-15.2 watchOS 2-8.3 DriverKit 19-21.2	iOS 10.3.1-15.2 tvOS 10.2-15.2 watchOS 3.2-8.3	Swift 4 Swift 4.2 Swift 5.5
Xcode 14.3.1	macOS Ventura 13	iOS 16.4 macOS 13.3 tvOS 16.4 watchOS 9.4 DriverKit 22.4	i386 x86_64 armv7k arm64 arm64e arm64_32	Xcode 13.1	macOS Big Sur 11.3	iOS 15 macOS 12 tvOS 15 watchOS 8 DriverKit 210.1	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15 iPadOS 13-15 macOS 10.9-12 tvOS 9-15 watchOS 2-8 DriverKit 19-210.1	iOS 10.3.1-15 tvOS 10.2-15 watchOS 3.2-8	Swift 4 Swift 4.2 Swift 5.5
Xcode 14.3*	macOS Ventura 13	iOS 16.4 macOS 13.3 tvOS 16.4 watchOS 9.4 DriverKit 22.4	i386 x86_64 armv7k arm64 arm64e arm64_32	Xcode 13	macOS Big Sur 11.3	iOS 15 macOS 11.3 tvOS 15 watchOS 8 DriverKit 20.4	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-15 iPadOS 13-15 macOS 10.9-11.3 tvOS 9-15 watchOS 2-8 DriverKit 19-20.4	iOS 10.3.1-15 tvOS 10.2-15 watchOS 3.2-8	Swift 4 Swift 4.2 Swift 5.5
Xcode 14.2	macOS Monterey 12.5	iOS 16.2 macOS 13.1 tvOS 16.1 watchOS 9.1 DriverKit 22.2	i386 x86_64 armv7k arm64 arm64e arm64_32	Xcode 12.5	macOS Big Sur 11	iOS 14.5 macOS 11.3 tvOS 14.5 watchOS 7.4 DriverKit 20.4	i386 x86_64 armv7 armv7s armv7k arm64 arm64e arm64_32	iOS 9-14.6 iPadOS 13-14.6 macOS 10.9-11.4 tvOS 9-14.6 watchOS 2-7.5 DriverKit 19-20.4	iOS 10.3.1-14.5 tvOS 10.2-14.5 watchOS 3.2-7.4	Swift 4 Swift 4.2 Swift 5.4



## ■ macOS 확인하기

- 왼쪽 최상단  를 클릭->이 맥에 관하여





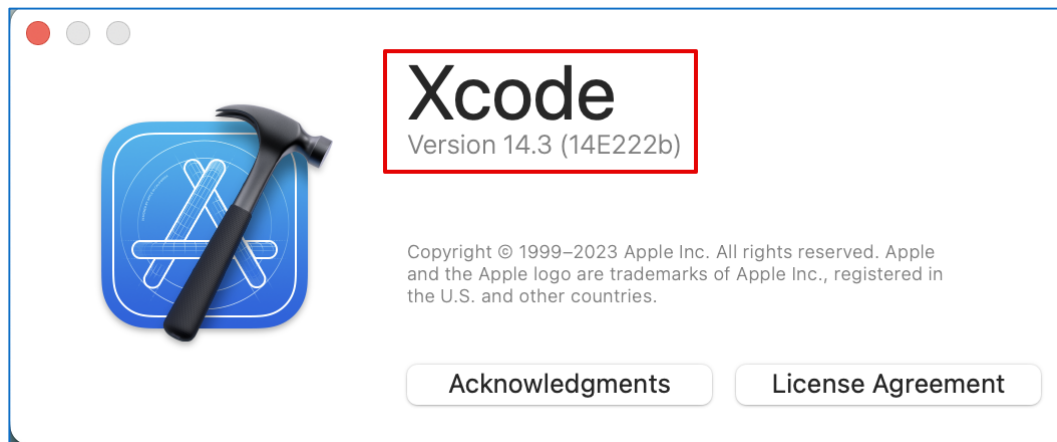


## ■ xcode 설치: App Store에서 무료로 설치

- 주의: 항상 최신 버전이 설치되므로 불필요하게 과중한 xcode 기능이 설치될수 있으므로 매우 주의하여야 함
- Old Version으로 돌아가기 위해서는 애플 개발자 사이트에서 다운 받아 설치하여야 함

## ■ xcode 실행

- LaunchPad나 Doc에서  을 찾아서 클릭
- 왼쪽 최상단 메뉴  오른쪽 xcode를 클릭 -> "about xcode" 클릭





# xcode 프로젝트 만들기

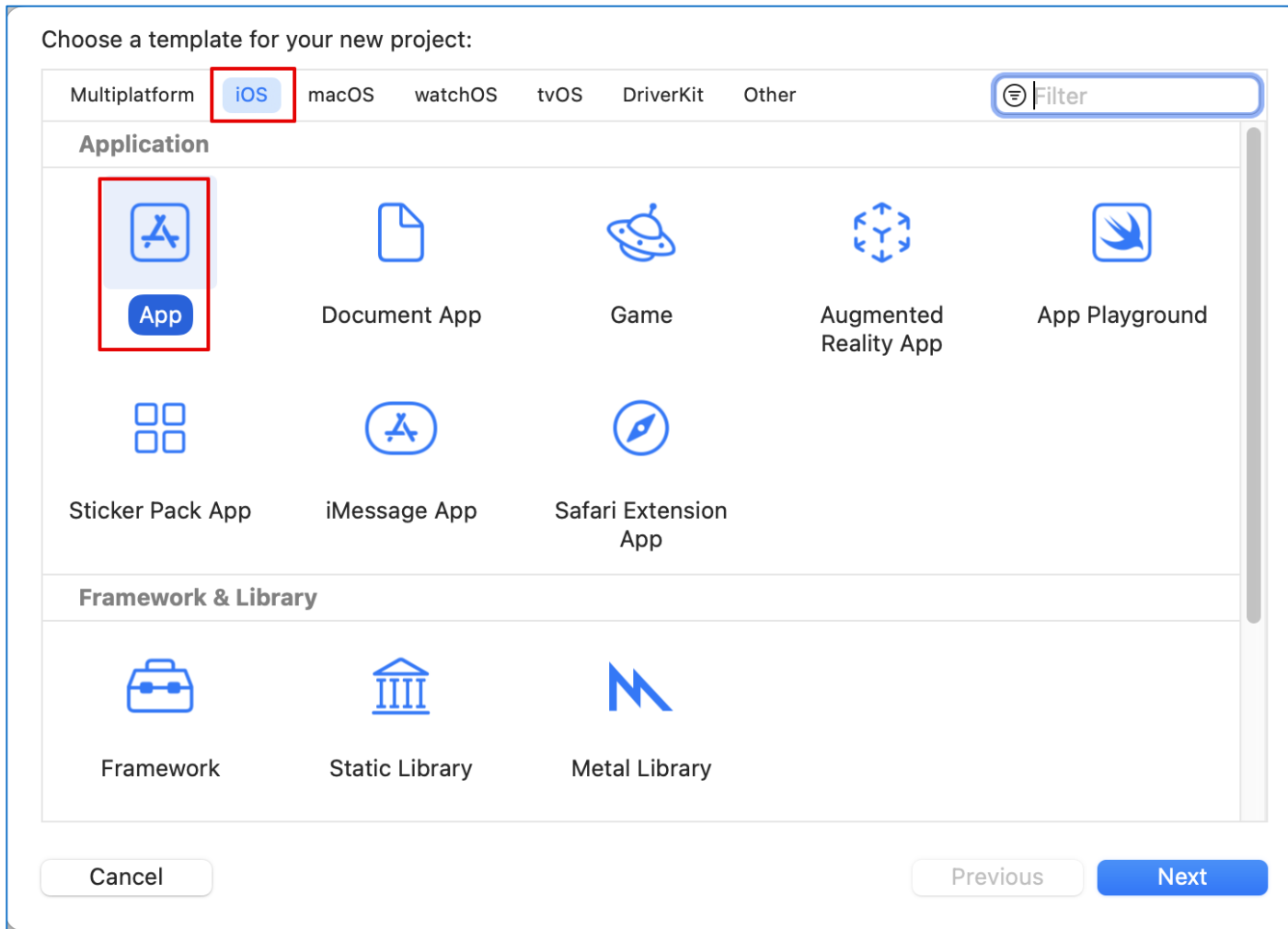
- xcode를 클릭하고 “Create a new Xcode project”클릭





# xcode 프로젝트 만들기

## Single View App 선택







# xcode 프로젝트 만들기

Choose options for your new project:

Product Name: SimpleApp

Team: Add account...

Organization Identifier: jmlee.hansung.ac.kr

Bundle Identifier: jmlee.hansung.ac.kr.SimpleApp

Interface: Storyboard

Language: Swift

Storage: None

☐ Host in CloudKit

☐ Include Tests

swiftUI  
Storyboard  
중에서  
Storyboard  
선택

반드시  
unchecked  
하라

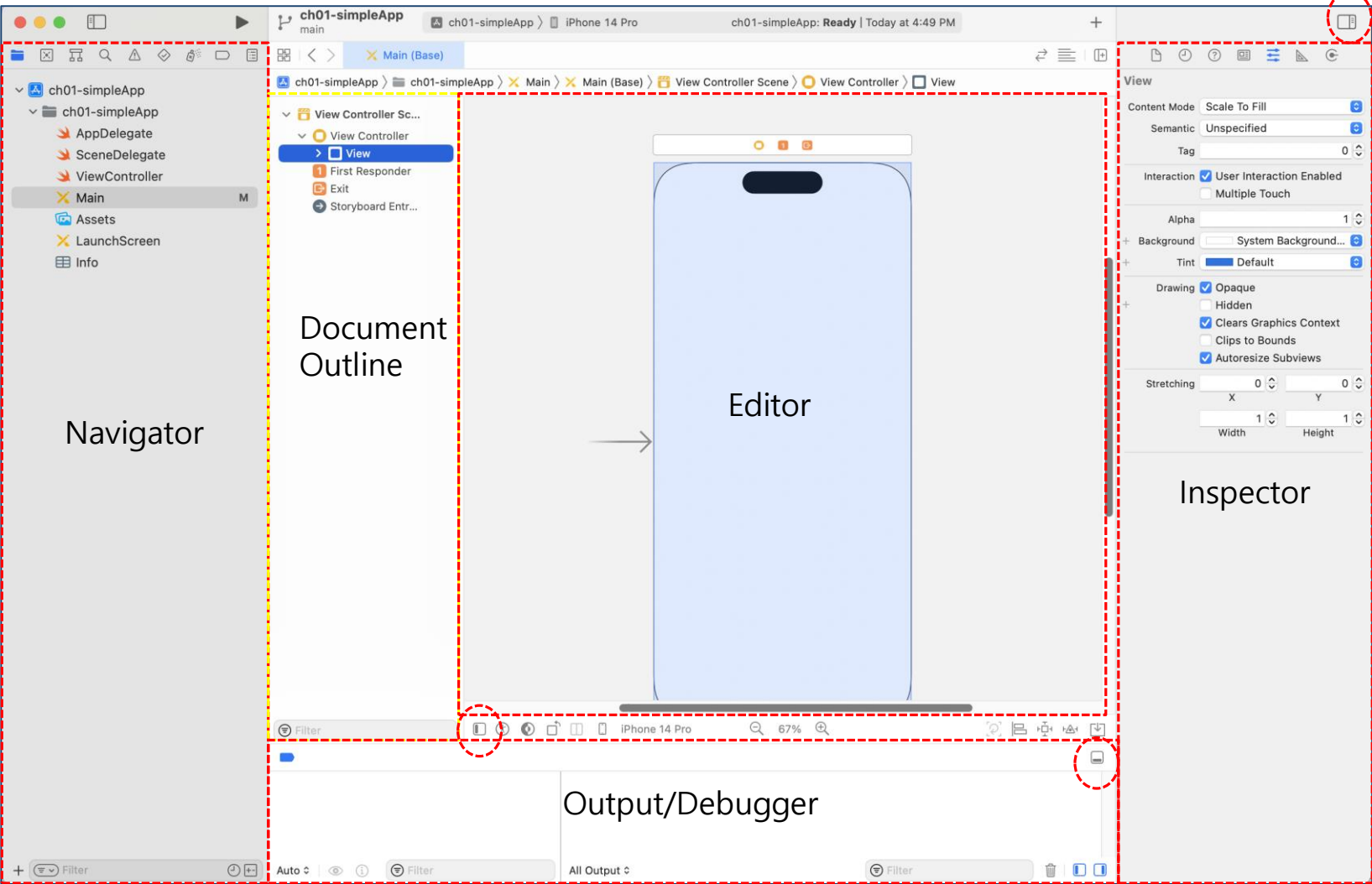
Cancel

Previous

Next



# xcode





- **Navigator**
  - 프로젝트 파일, 테스트 등 프로젝트에 포함되는 입력
  - 디버깅 정보
  - 버전 컨트롤 등
- **Editor**
  - 소스코드
  - 스토리보드를 이용한 UI 에디팅
- **Inspector**
  - Editor에서 선택된 오브젝트에 대한 속성 등
- **Output**
  - 실행 시 발생하는 출력
  - 브레이크 포인트에서 각종 변수 등의 값

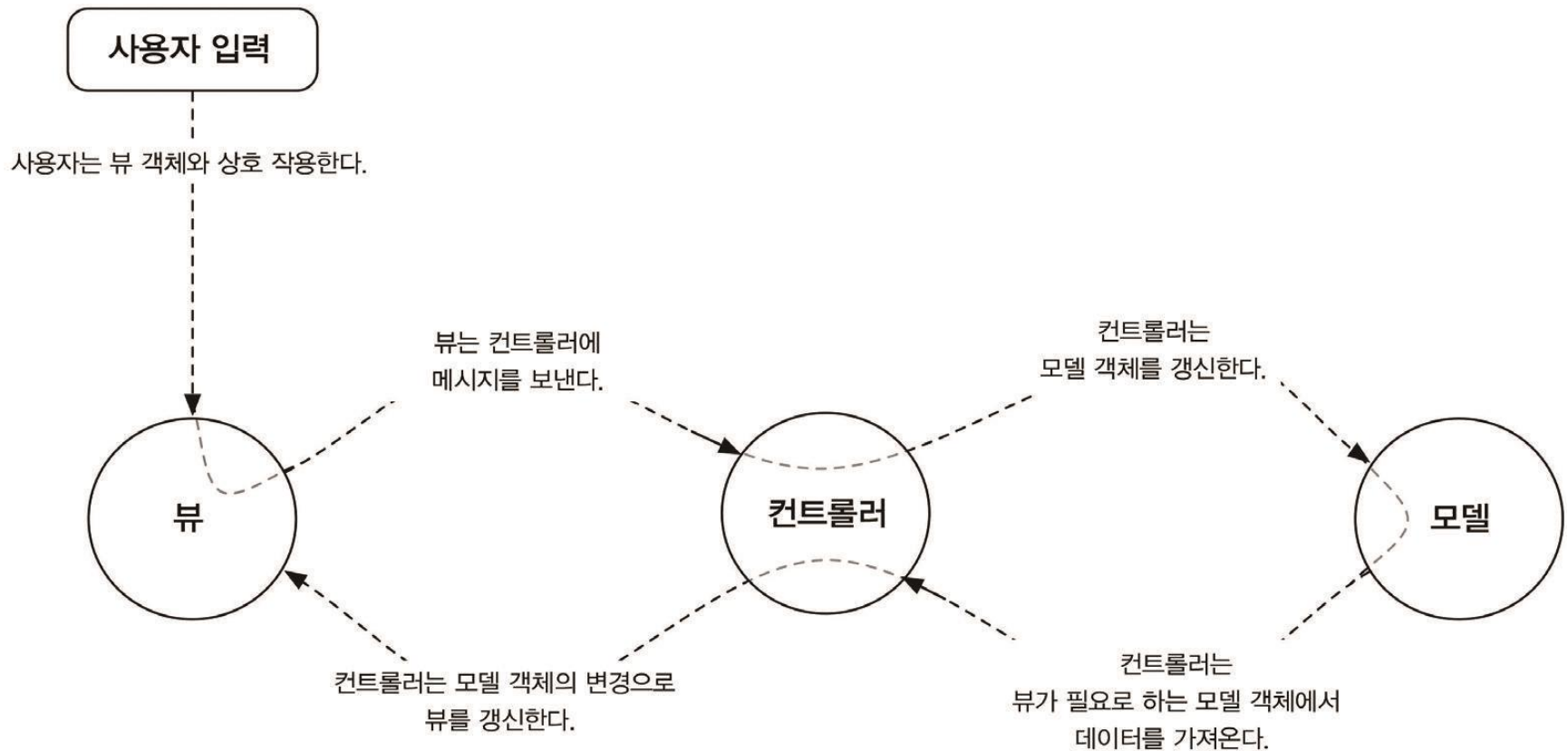


# 모델-뷰-컨트롤러(MVC)

- **Model-View-Controller**은 iOS 개발에 사용되는 핵심 디자인 패턴
  - MVC에서 각 객체는 모델, 뷰, 컨트롤러 레이어 중의 하나
  - 모델 레이어
    - 데이터를 저장하지만 사용자 인터페이스에 대하여 알지 못함
    - simpleApp앱에서 모델은 문제와 정답의 목록을 가지는 문자열 리스트로 구성
    - 대개 모델 레이어는 현실 세계의 사물을 대상으로 함
  - 뷰 레이어
    - 사용자에게 보이는 모든 객체를 포함
    - 버튼, 텍스트필드, 슬라이더 등이 있음
    - 대부분 이미 만들어져 있는 것을 그대로 사용함
  - 컨트롤러 레이어
    - 응용 프로그램을 관리
    - 사용자에게 보이는 뷰를 구성
    - 뷰와 모델을 연결하고 동기화 함



# MVC 패턴



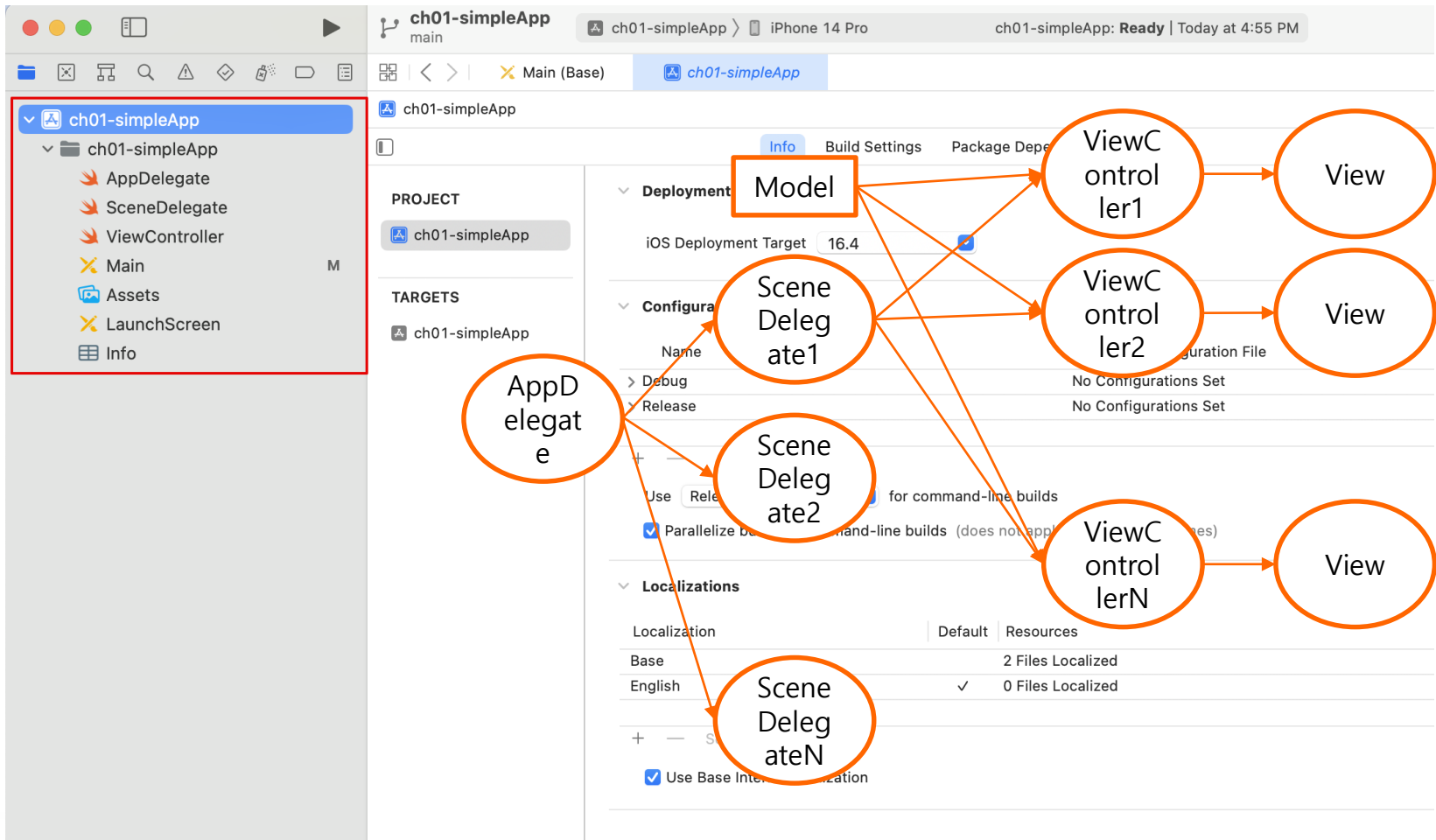
모델과 뷰는 직접적으로 통신하지 않는다. 둘은 항상 분리 가능하다.



# simpleApp 앱

## ■ 기본적으로 생성된 프로젝트의 구성

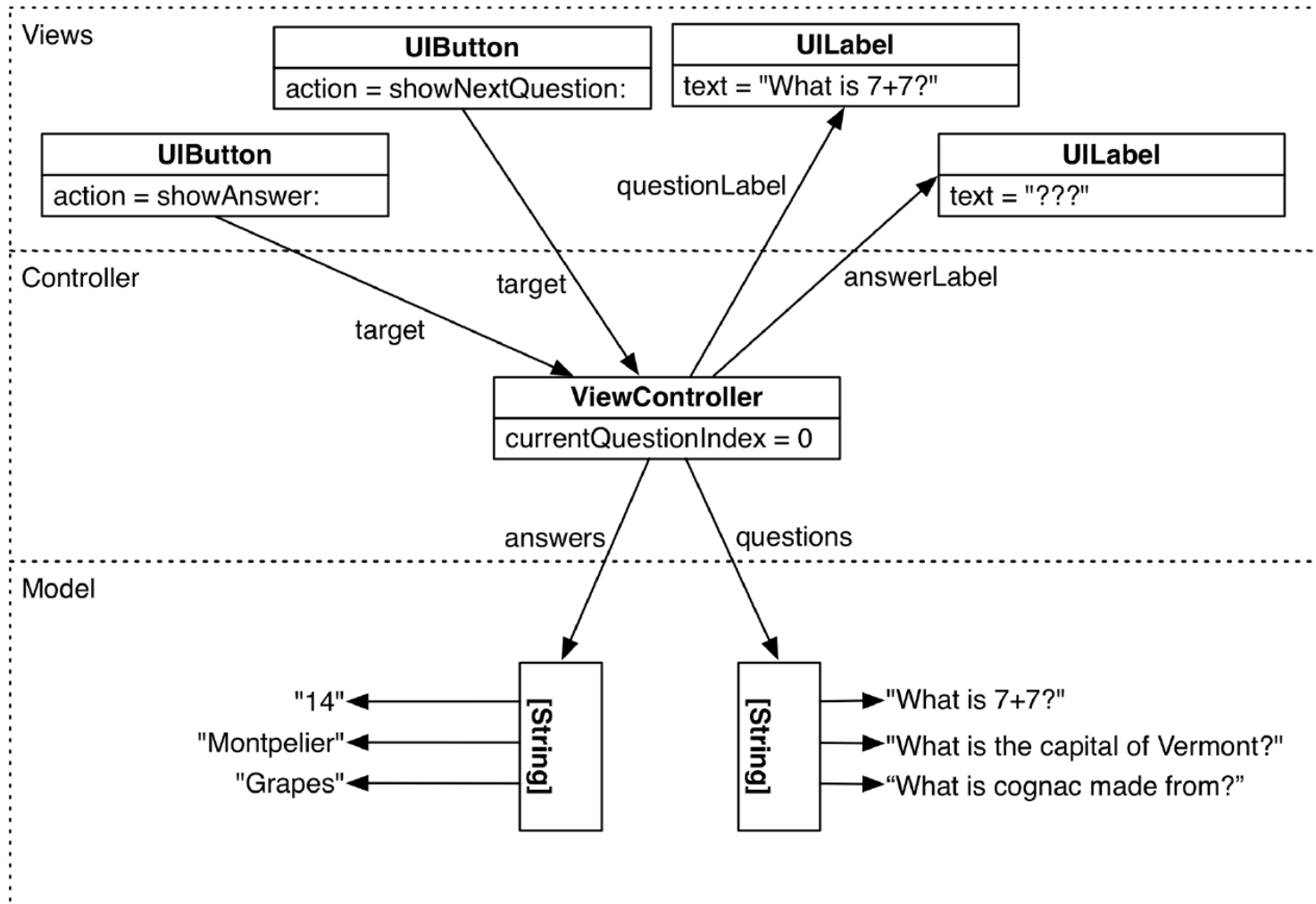
- iOS 13이후 부터 multiple Scene를 지원
- 이것은 점차적으로 단말기가 커져가는 것에 대비하기 위함





# simpleApp 앱

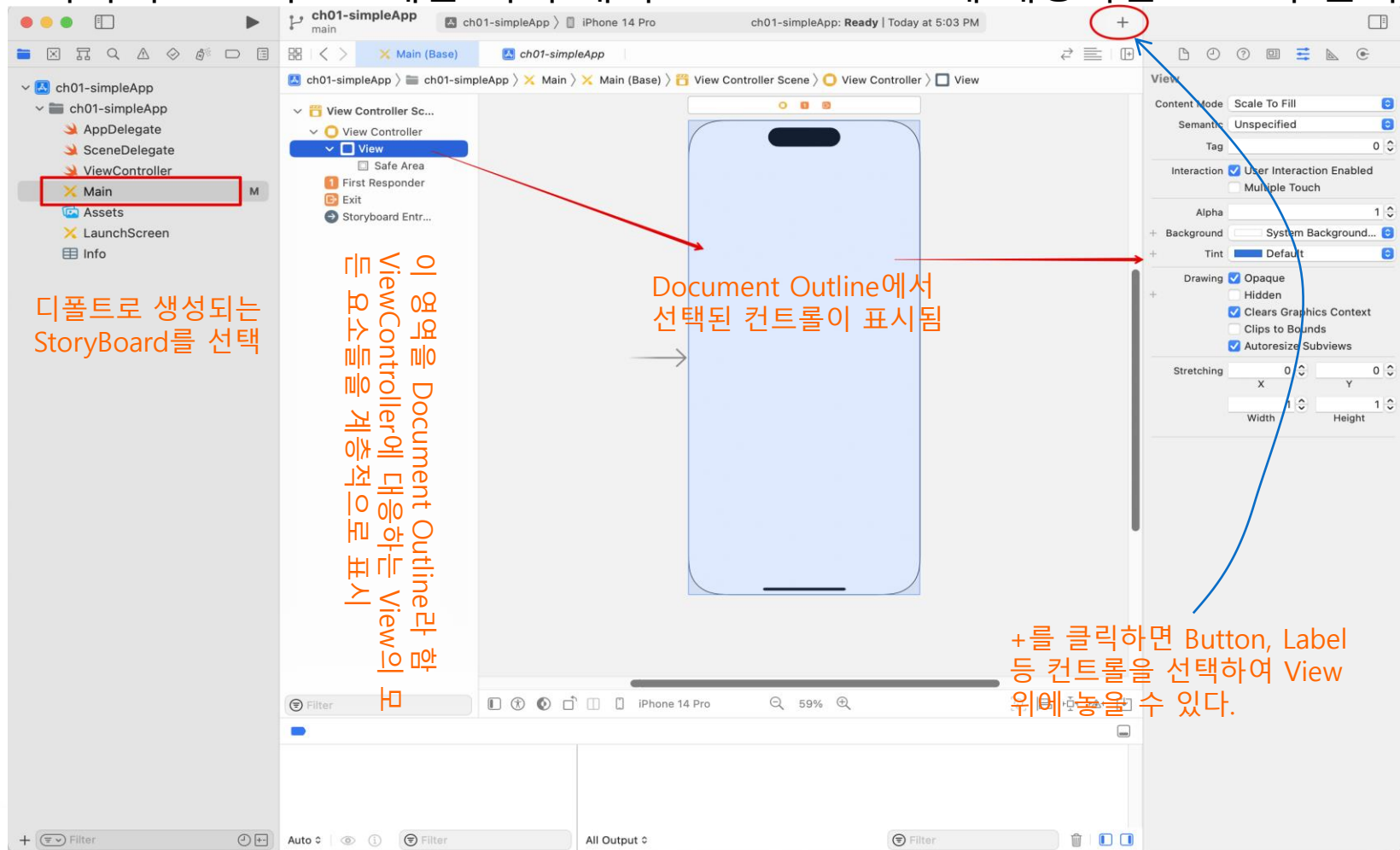
- 모델 레이어: 두개의 [String] 인스턴스로 구성
- 뷰 레이어: 두개의 UILabel 인스턴스와 두개의 UIButton 인스턴스로 구성
- 컨트롤러 레이어: ViewController 인스턴스로 구성





# View 디자인

- StoryBoard: 안드로이드에서 layout의 xml과 동일
  - StoryBoard를 이용하여 뷰를 디자인함
  - 하나의 스토리보드에는 여러 개의 ViewController에 해당하는 View가 들어감







# Object Library

- View위에 드래그하여 놓을 수 있는 컨트롤들

를 클릭하면 아래와 같이 Object Library가 나옴

검색단어를 입력할 수 있음

원하는 컨트롤을 선택하여 View위로 드래그&드롭을 하면 View에 나타남

**Button**  
UIButton: Sends an action when tapped

You can set the title, image, and other appearance properties of a button. In addition, you can specify a different appearance for each button state.

1 2 Segmented Control

Text Text Field

Slider

Switch

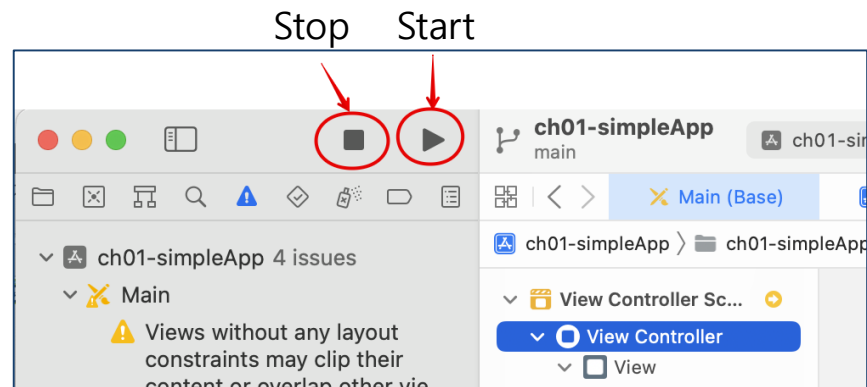
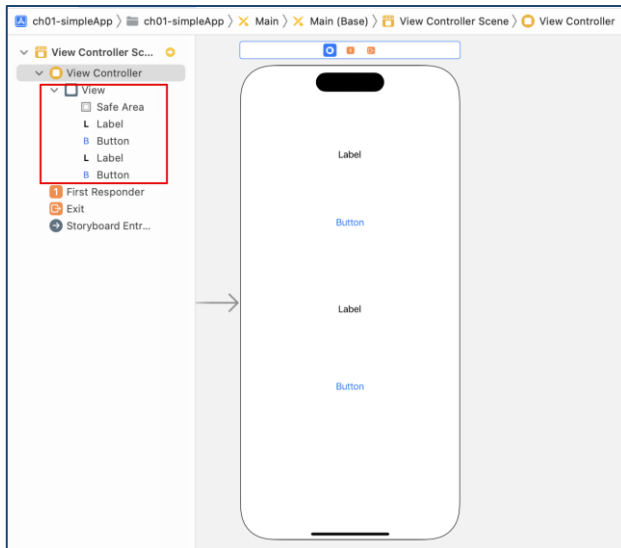
Activity Indicator View



# simpleApp 앱 디자인

## ■ ObjectLibrary를 이용한 UI 디자인

- Main.storyboard를 선택하라
- 우측상단의 Object Library(+)를 클릭하라
- Label 컨트롤을 선택하여 View로 드래그&드롭하라
- 동일한 방법으로 Button, Label, Button을 View로 드래그&드롭하라
- 적절히 정렬하라
- Document Outline을 우측 그림과 같은지 확인하라.
- 좌측 상단의 삼각형을 클릭하여 실행하여 보라





# 간단한 속성 변경

## Inspector에서 5번째 탭 선택

ning ch01-quizz on iPhone SE (2nd generation)

... > ... > View > L Question Label < > > > >

Label

Text Plain (2) 여기서 원하는 텍스트를 입력한다.

Question

Color Default (Label C...)

Font System 17.0

Dynamic Type Automatically Adjusts Font

Alignment

Lines 1

Behavior ☒ Enabled ☐ Highlighted

Baseline Align Baselines

Line Break Truncate Tail

Autoshrink Fixed Font Size

☐ Tighten Letter Spacing

Highlighted Default

Shadow Default

Shadow Offset 0 -1

Width Height

View

Content Mode Left

(1) 임의의 컨트롤을 선택한다.

(1),(2)들 동시에 하기 위해서 더블클릭한후 직접입력하여도 된다.



# View와 UIViewController 연결

## ■ View와 UIViewController의 연결이란?

- UIViewController에서 Label에 데이터를 write하려고 하면 Label을 액세스할 수 있어야 한다.
  - 안드로이드에서 label = findViewById(...)와 동일
- 사용자의 View를 통하여 버튼을 클릭하면 그 내용은 UIViewController에 알려져야 한다.
  - 안드로이드에서는 button.setOnClickListener(...)와 동일

## ■ 연결하는 가장 쉬운 방법

- StoryBoard와 UIViewController을 동시에 오픈한다.
- StoryBoard에서 원하는 컨트롤을 "Control+드래그"하여 UIViewController의 클래스내에 드롭한다.
- Outlet/Action을 선택한다.
- 적절한 이름을 정한다.

### 동시 오픈 방법

1. Main.storyboard를 오픈한다
2. 이 상태에서 option 및 ViewController.swift를 클릭한다



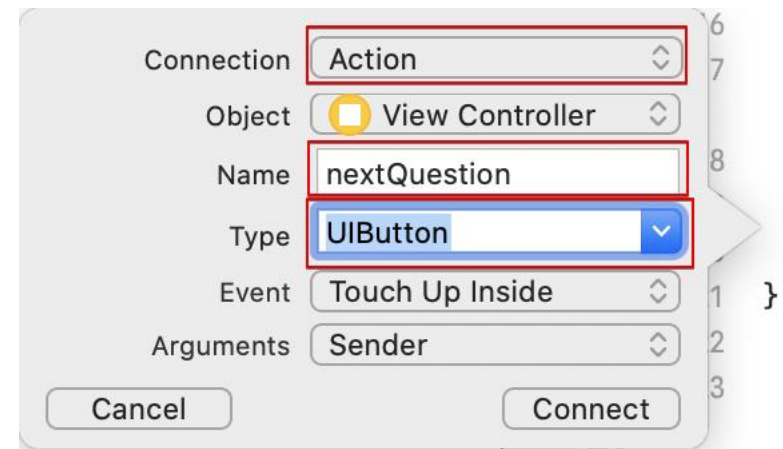
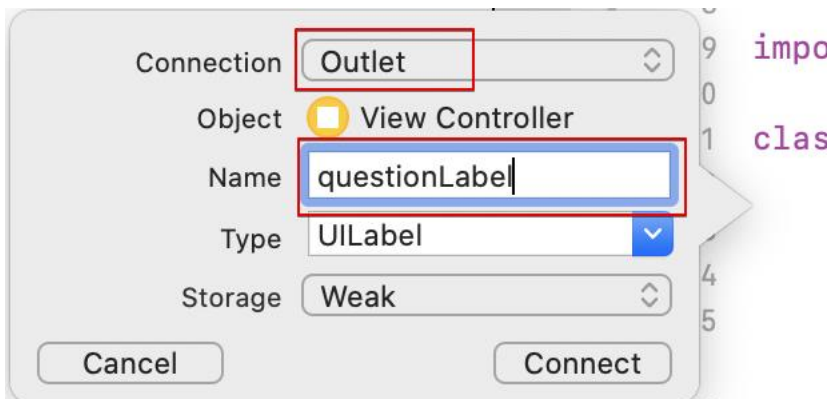
# View와 UIViewController 연결

## ■ Outlet 연결

- Outlet: 객체의 레퍼런스를 의미한다.
- 레이블 Question → Outlet: questionLabel
- 레이블 Answer → Outlet: answerLabel

## ■ Action 연결

- Action: 객체의 listener를 의미한다.
- 버튼 Next Question → Action: nextQuestion
- 버튼 Show Question → Action: showAnswer



# View와 UIViewController 연결

## ■ 결과

- 이것은 View의 어떤 컨트롤과 연결되었음을 의미한다.  
- 이것이 비어 있는 경우 연결되었다가 끊어졌다는 의미이고 에러의 원인이 된다.

```
1 //
2 // ViewController.swift
3 // ch01-quiz
4 //
5 // Created by Jae Moon Lee on 2021/01/01.
6 // Copyright © 2021 Jae Moon Lee. All rights reserved.
7 //
8
9 import UIKit
10
11 class ViewController: UIViewController {
12
13     @IBOutlet weak var questionLabel: UILabel!
14     @IBOutlet weak var answerLabel: UILabel!
15     override func viewDidLoad() {
16         super.viewDidLoad()
17         // Do any additional setup after loading
18         // the view.
19     }
20
21     @IBAction func nextQuestion(_ sender: UIButton)
22     {
23
24     }
25
26     @IBAction func showAnswer(_ sender: UIButton) {
27     }
28 }
```

UI Elements: Question, Next Question, Answer, Show Answer

Code Elements: @IBOutlet weak var questionLabel: UILabel!, @IBOutlet weak var answerLabel: UILabel!, override func viewDidLoad() { super.viewDidLoad() }, @IBAction func nextQuestion(\_ sender: UIButton) { }, @IBAction func showAnswer(\_ sender: UIButton) { }

UIViewController에서 questionLabel에 임의의 문자를 쓰면 View의 Question에 나타나고 View에서 "Next Question"을 클릭하면 UIViewController의 함수 nextQuestion()이 호출됨을 의미



# 연결 확인

## ■ Connection Inspector

(2) Connection Inspector 선택

The screenshot shows the Xcode interface with the Connection Inspector on the right. The storyboard on the left contains a 'View Controller' with a 'View' containing a 'Safe Area' and several UI elements: 'Question Label', 'Next Question', 'Answer Label', 'Show Answer', 'First Responder', and 'Exit'. The Connection Inspector on the right is divided into several sections: 'Triggered Segues' (manual), 'Outlets' (answerLabel, questionLabel, searchDisplayController, view), 'Presenting Segues' (Relationship, Show, Show Detail, Present Modally, Present As Popover, Embed, Push (deprecated), Modal (deprecated), Custom), 'Referencing Outlets' (New Referencing Outlet), 'Referencing Outlet Collections' (New Referencing Outlet Collection), and 'Received Actions' (nextQuestion:, showAnswer:). Red lines connect the 'Question Label' to 'questionLabel', 'Next Question' to 'nextQuestion:', 'Answer Label' to 'answerLabel', and 'Show Answer' to 'showAnswer:'. A blue arrow points to the 'Received Actions' section.

(1) ViewController을 선택

X를 선택하면 연결이 끊어 짐



# 모델 레이어 만들기

- 모델이 너무 간단하기에 Controller Layer에 포함

```
// ViewController.swift
class ViewController: UIViewController {

    var questions = [
        "대한민국의 수도는 무엇인가요?",
        "한국 청년들에게 가장 인기있는 대학은 무슨 대학인가요?",
        "7+21은 얼마인가요?"
    ]
    var answers = [
        "서울",
        "한성대학교",
        "28"
    ]
    var currentIndex = 0

    @IBOutlet weak var questionLabel: UILabel!
    @IBOutlet weak var answerLabel: UILabel!

    ...

}
```





## ■ viewDidLoad() 함수

- 이 함수는 프로그램이 완전히 실행된 직후 가장 처음 호출되는 Callback 함수이다.
- 이 함수에서 이 View에 대한 초기화를 한다
  - 주의: 생성자는 아님

```
// ViewController.swift

override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view.
    questionLabel.text = questions[currentIndex]
}
```



## ■ nextQuestion/showAnswer 함수 구현

```
// ViewController.swift

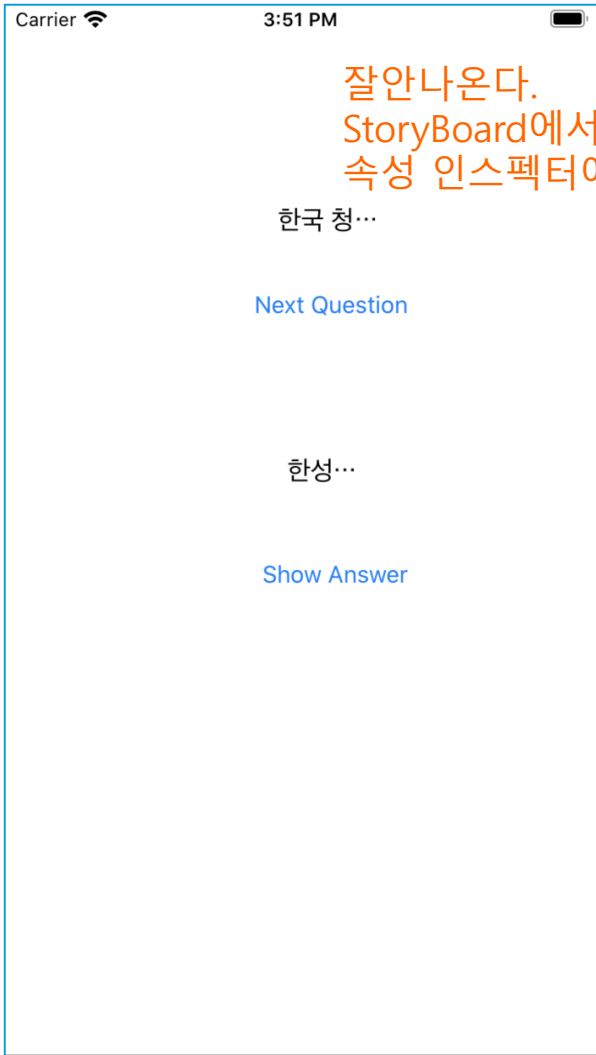
@IBAction func nextQuestion(_ sender: UIButton) {
    currentIndex = (currentIndex+1)%questions.count
    questionLabel.text = questions[currentIndex]
}

@IBAction func showAnswer(_ sender: UIButton) {
    answerLabel.text = answers[currentIndex]
}
```



# 실행 그리고 수정

## ■ 실행 결과



잘안나온다.  
Storyboard에서 크기를 적절히 변경하라.  
속성 인스펙터에서 중앙에 오도록 하라.

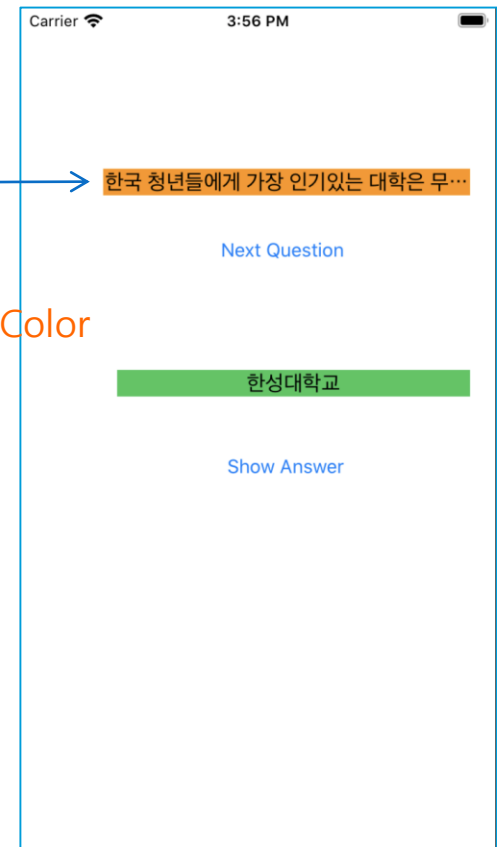
한국 청...

Next Question

한성...

Show Answer

Alignment: Center  
Background: System Orange Color



한국 청년들에게 가장 인기있는 대학은 무...

Next Question

한성대학교

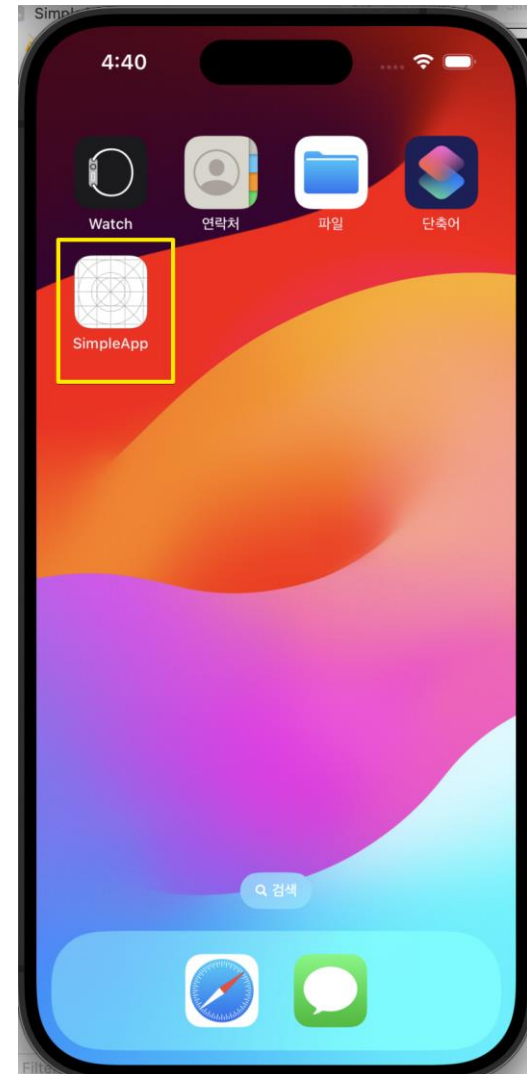
Show Answer



# 앱 아이콘

## ■ 앱을 보여주는 아이콘

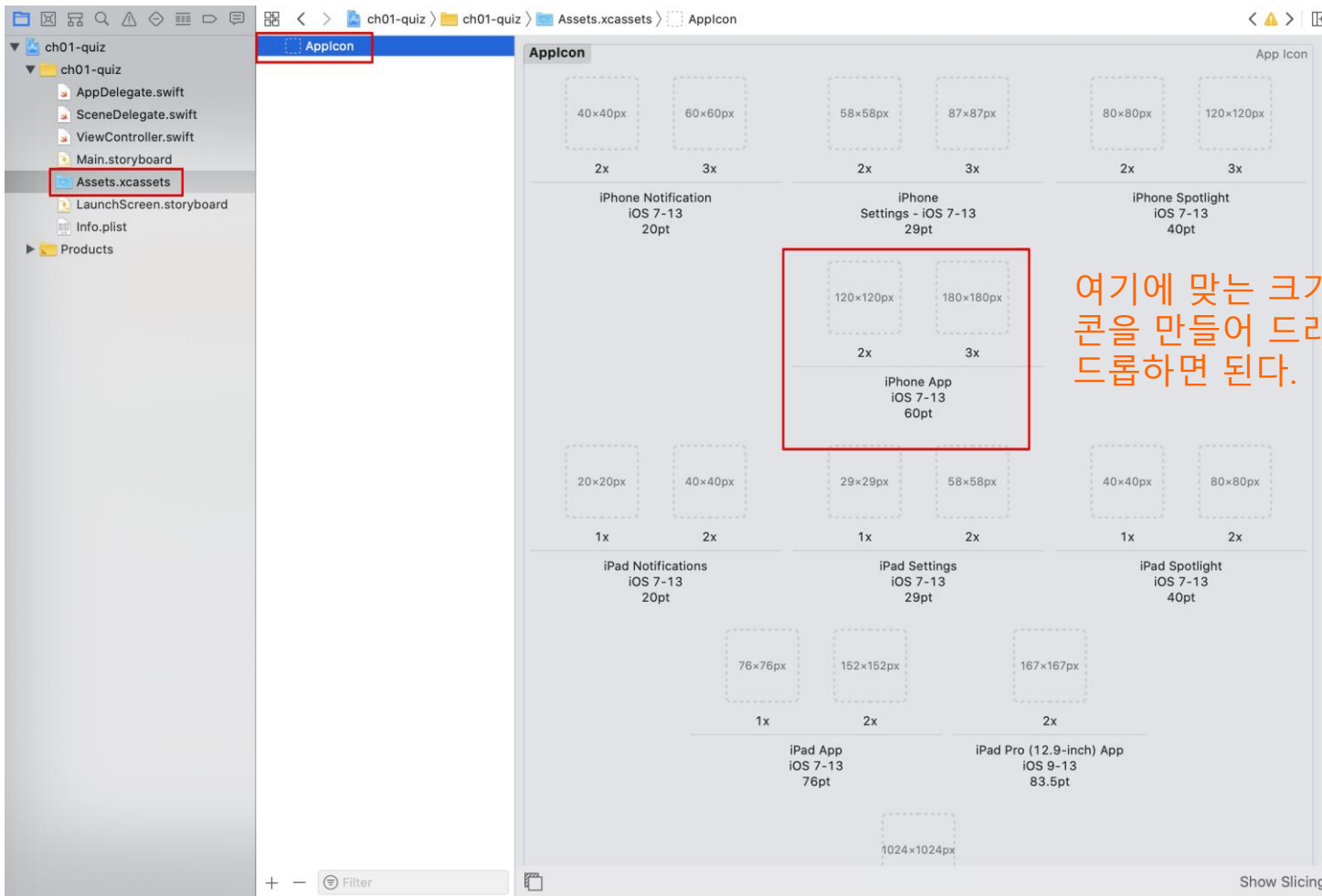
- 현재
- 원하는 이미지로 바꿀수 있다.





# 앱 아이콘

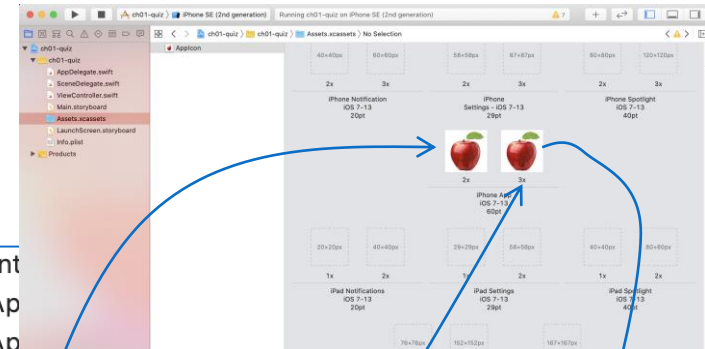
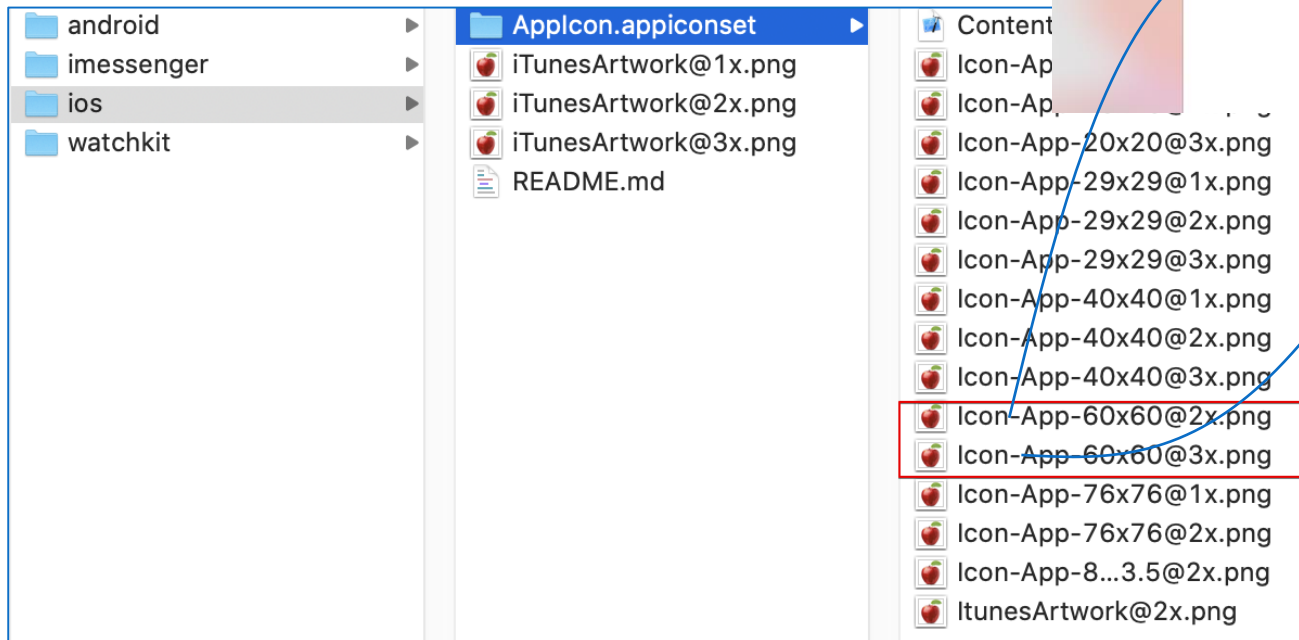
- iOS는 기기별 아이콘의 크기를 정해 놓음
  - 여러 기기에 적용하기 위해서는 적용을 원하는 기기에 해당하는 아이콘을 전부 준비하여야 함



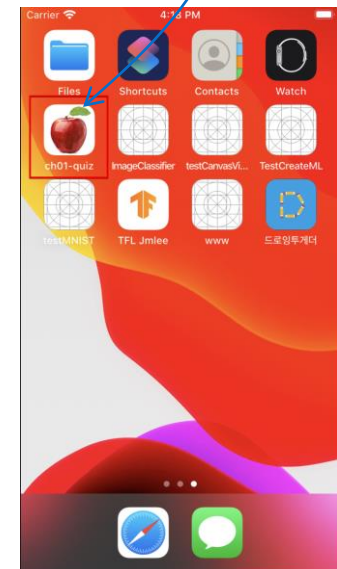


# 앱 아이콘

- 아이콘 자동으로 만들어주는 사이트
  - 인터넷에 엄청 많다
  - <https://makeappicon.com/>
    - 이 사이트에서 만든 아이콘



Drag & drop

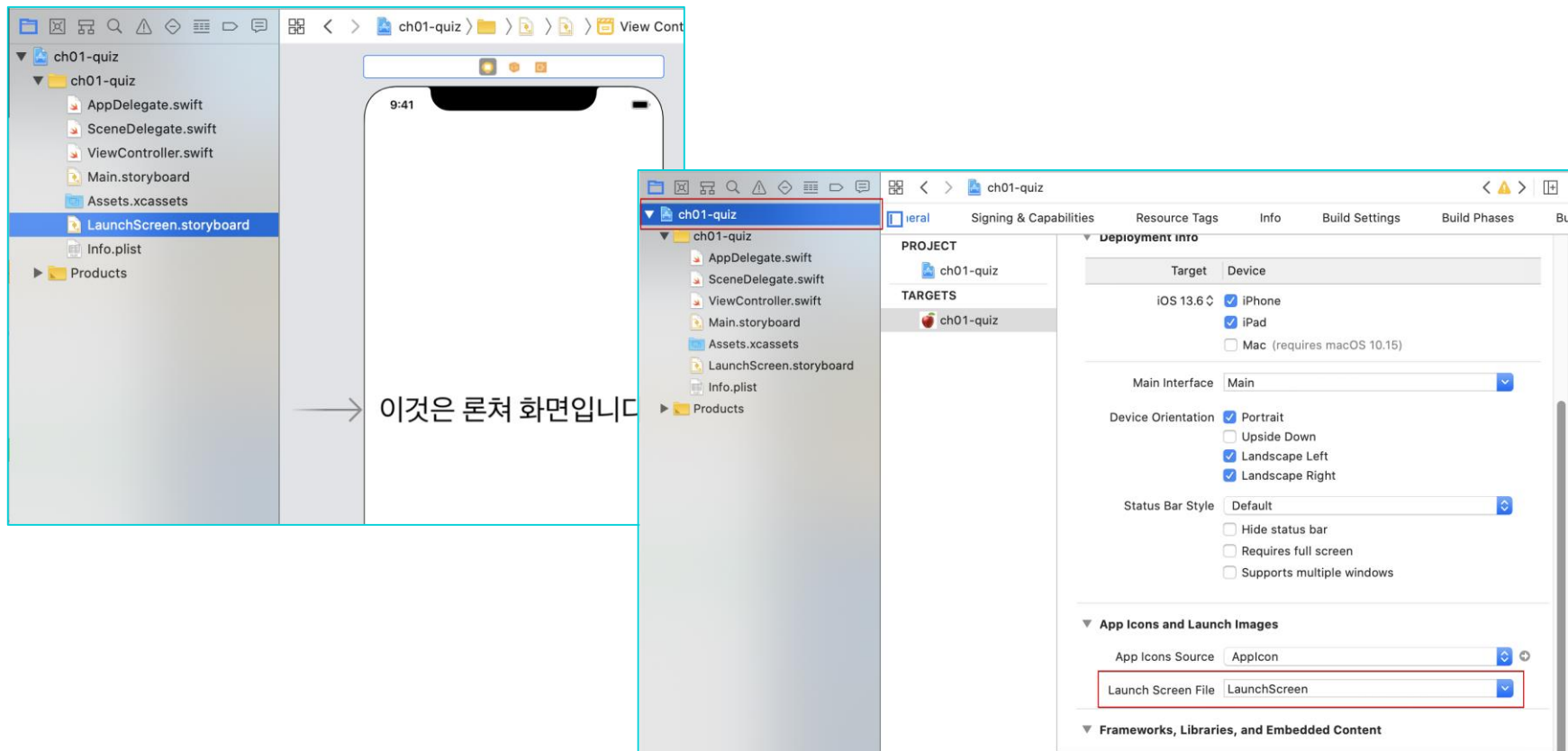




# 론처 화면

## ■ 앱이 로딩 될때 나타나는 화면

- LaunchScreen.storyboard를 꾸민다.
- Ch01-simpleApp → General → App Icons & Launch Images → Launch Screen File에서 LaunchScreen 선택

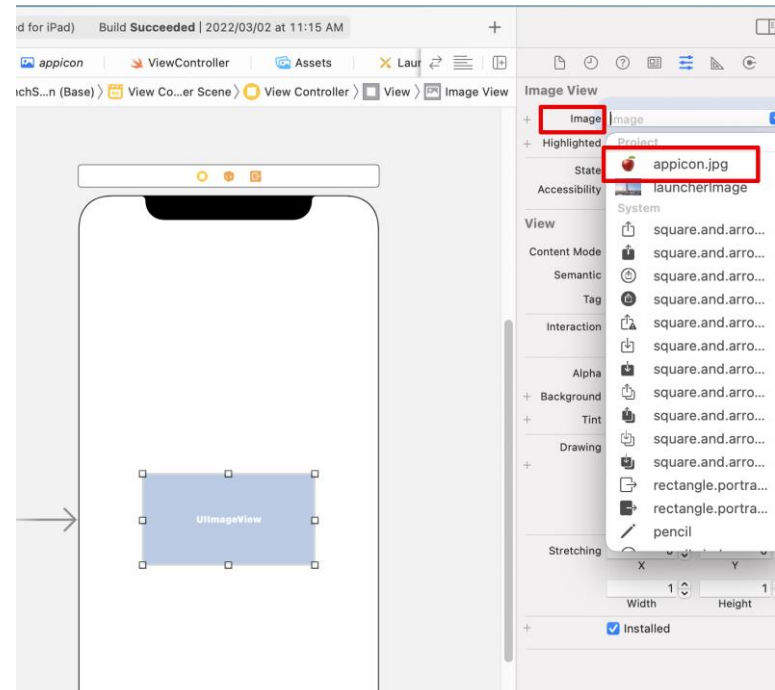
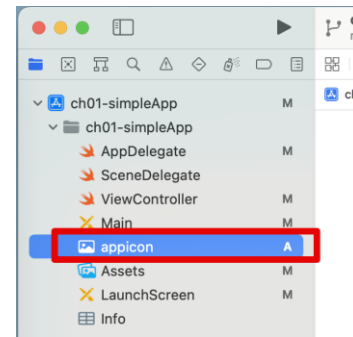




# 론처 화면

## ■ 론처 화면 꾸미기

- appleicon.jpg를 Navigator에 복사한다.
- LaunchScreen.storyboard를 open한다
- Object Library에서 Image 컨트롤을 drag하여 LaunchScreen.storyboard 중앙에 적절히 놓는다.
- Image컨트롤을 선택하여 속성 창에서 appleicon.jpg를 선택한다.







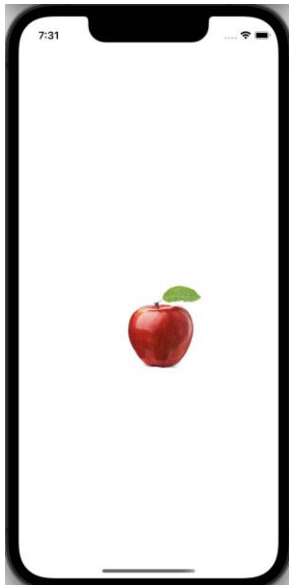
# 론처 화면

## ■ 론칭 시간 제어

- Navigator에서 AppDelegate.swift를 선택하여 오픈하여 아래 코드를 삽입하라.

```
func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UI/  
    // Override point for customization after application launch.  
    Thread.sleep(forTimeInterval: 2.0)  
    return true  
}
```

## ■ 실행



2초후

