

# iOS 开发入门

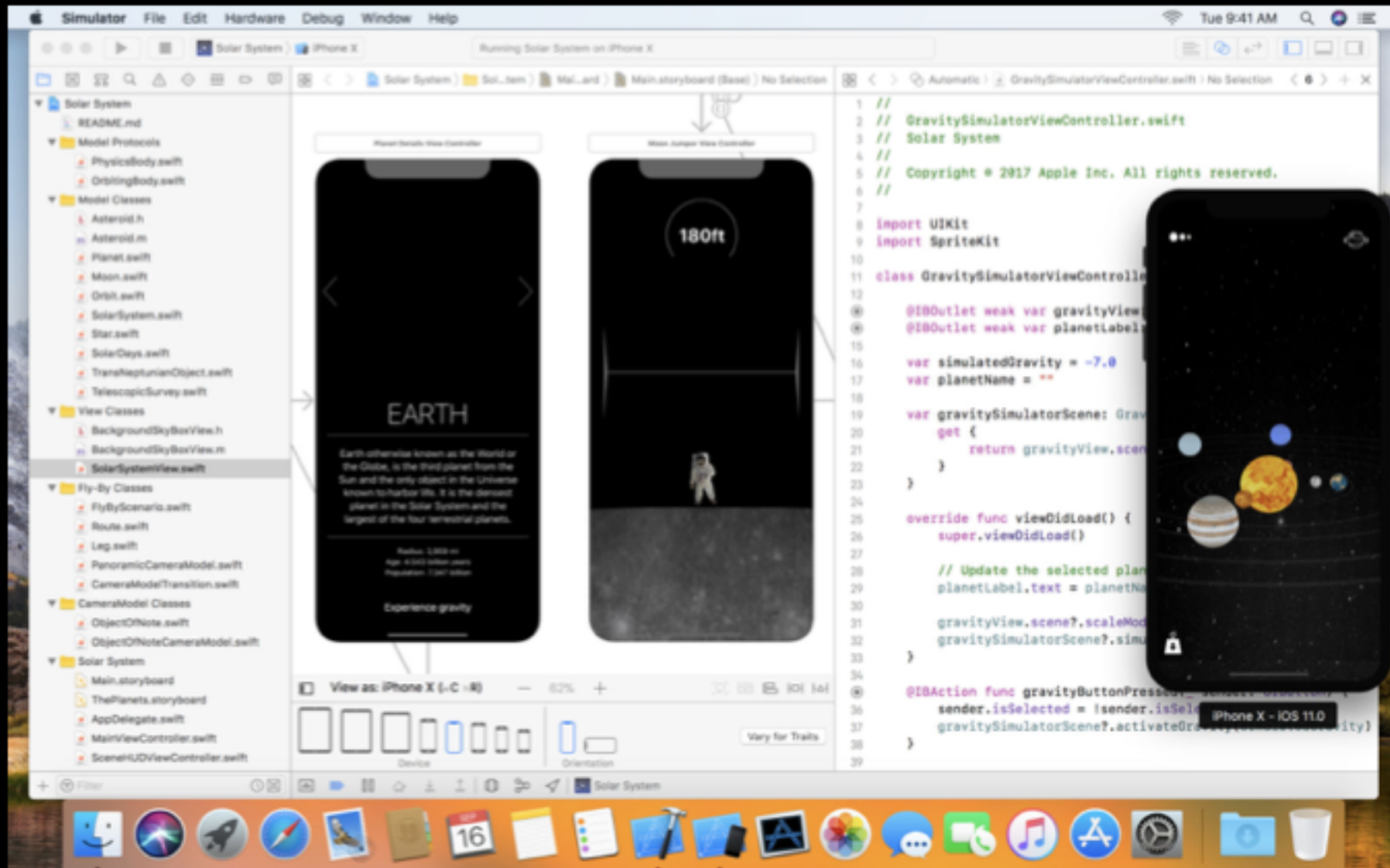
# iOS 开发入门

- 开发环境 macOS + Xcode
- 开发组件
- 语言 Swift
- 架构模式 MVC
- Demo: 开发一个简单的计算器

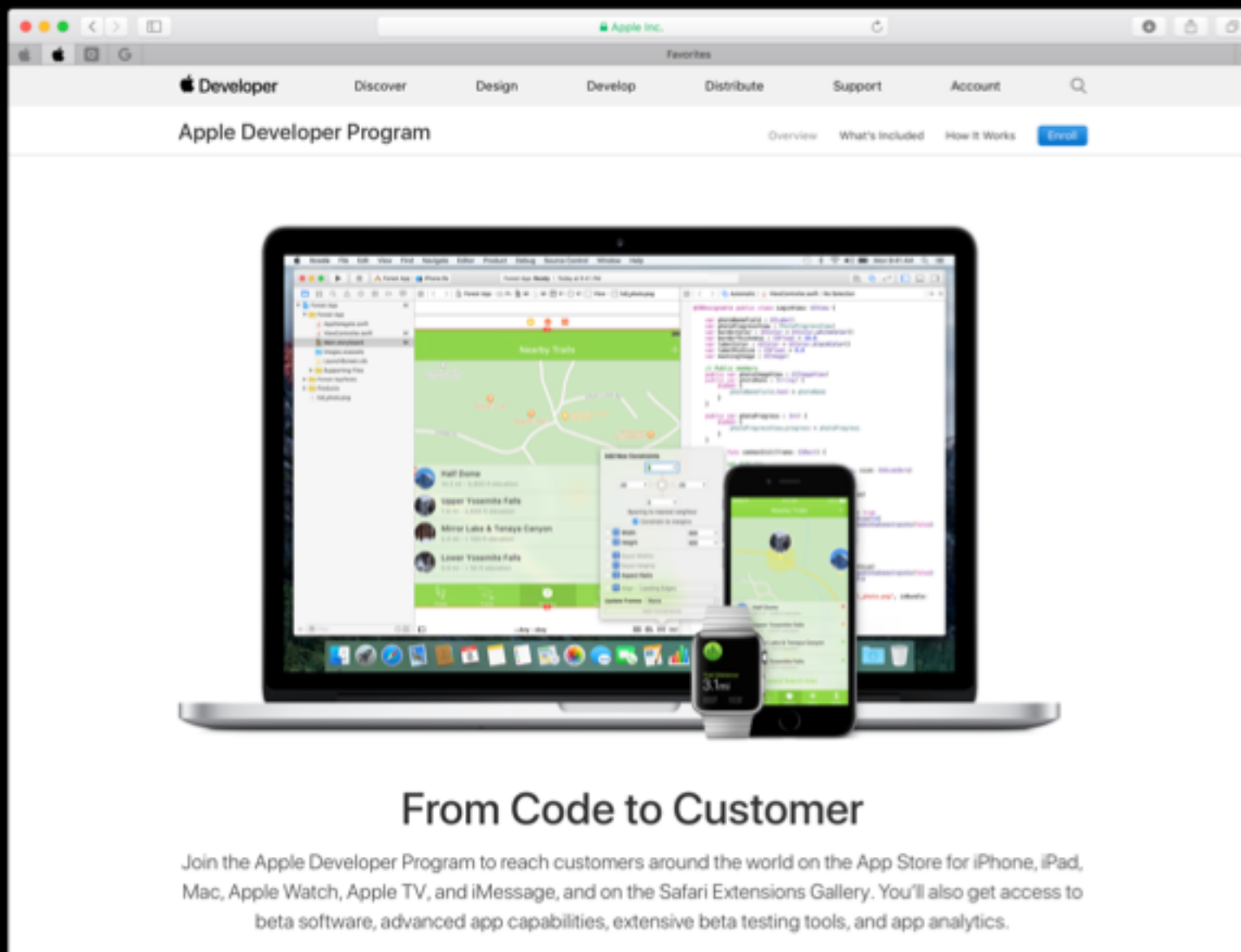
# 开发环境 macOS + Xcode



# 开发环境 macOS + Xcode



# 开发者账号



The image is a screenshot of the Apple Developer Program website, displayed within a browser window. The browser's address bar shows 'Apple Inc.' and the page title is 'Apple Developer Program'. The navigation bar includes links for 'Developer', 'Discover', 'Design', 'Develop', 'Distribute', 'Support', and 'Account'. Below the navigation bar, the main heading 'Apple Developer Program' is followed by links for 'Overview', 'What's Included', 'How It Works', and a blue 'Enroll' button. The central visual is a composite image featuring a laptop, a smartphone, and a smartwatch. The laptop screen displays a map application with a sidebar listing locations like 'Half Dome', 'Upper Yosemite Falls', 'Mirror Lake & Tenaya Canyon', and 'Lower Yosemite Falls'. The smartphone and smartwatch also show the same map application. The smartwatch screen displays '3.1 mi' and '100%'.

## From Code to Customer

Join the Apple Developer Program to reach customers around the world on the App Store for iPhone, iPad, Mac, Apple Watch, Apple TV, and iMessage, and on the Safari Extensions Gallery. You'll also get access to beta software, advanced app capabilities, extensive beta testing tools, and app analytics.

# 开发组件

- Core OS: Keychain Access, Certificates, Security ...
- Core Services: Core Location, Core Data, Core Motion, Core ML, GCD ...
- Media: Core Graphics, Core Animation, Core Audio ...
- Cocoa Touch: UIKit, MapKit, EventKit, SceneKit, SpriteKit, ARKit, CloudKit, Photos

**Swift or Objective-C?**

# Swift

- 安全 (强类型, 可选值, ARC)
- 运行快
- 表达型

```
myArray.insert(item, at: 0)
```



# Swift 常量 & 变量

```
var myVariable = 42
```

```
myVariable = 50
```

```
let myConstant = 42
```

```
let implicitInteger = 70
```

```
let implicitDouble = 70.0
```

```
let explicitDouble: Double = 70
```

```
var variableArray = [1, 2, 3]
```

```
variableArray[0] = 4
```

```
let constantArray = [1, 2, 3]
```

```
constantArray[0] = 4 // Error!
```

# Swift 控制流

```
let names: [String?] = ["John", "Mike", "Alice", nil]
for name in names {
    if let name = name {
        print(name)
    }
}
```

```
for name in names {
    guard let name = name else { continue }
    print(name)
}
```

```
for name in names where name != nil {
    print(name!)
}
```

# Swift 函数 & 闭包

```
func calculateStatistics(scores: [Int]) -> (min: Int, max: Int, sum: Int) {  
    ...  
}
```

```
let calculateStatistics: (scores: [Int]) -> (min: Int, max: Int, sum: Int) = {  
    ...  
}
```

```
let calculateStatistics = { (scores: [Int]) -> (min: Int, max: Int, sum: Int) in  
    ...  
}
```

```
session.downloadTask(with: url, completionHandler: { (url, response, error) in  
    ...  
})
```

# Swift Struct & Class

```
class Room {  
    var temperature: Double  
  
    init(temperature: Double) {  
        self.temperature = temperature  
    }  
  
    func raiseTemperature(by delta: Double) {  
        temperature += delta  
    }  
}
```

```
let room1 = Room(temperature: 24)  
let room2 = room1  
room1.raiseTemperature(by: 6)  
room1.temperature // 30  
room2.temperature // 30
```

```
struct Room {  
    var temperature: Double  
  
    mutating func raiseTemperature(by delta: Double) {  
        temperature += delta  
    }  
}
```

```
var room1 = Room(temperature: 24)  
let room2 = room1  
room1.raiseTemperature(by: 6)  
room1.temperature // 30  
room2.temperature // 24  
room2.raiseTemperature(by: 4) // Error!
```

# Swift 闭包

```
let array = [1, 2, 3]
let sortedArray = array.sorted(by: { (first, second) -> Bool in
    return first > second
})
```

```
let sortedArray = array.sorted(by: { (first, second) in
    return first > second
})
```

```
let sortedArray = array.sorted(by: { (first, second) in
    first > second
})
```

```
let sortedArray = array.sorted(by: { $0 > $1 })
```

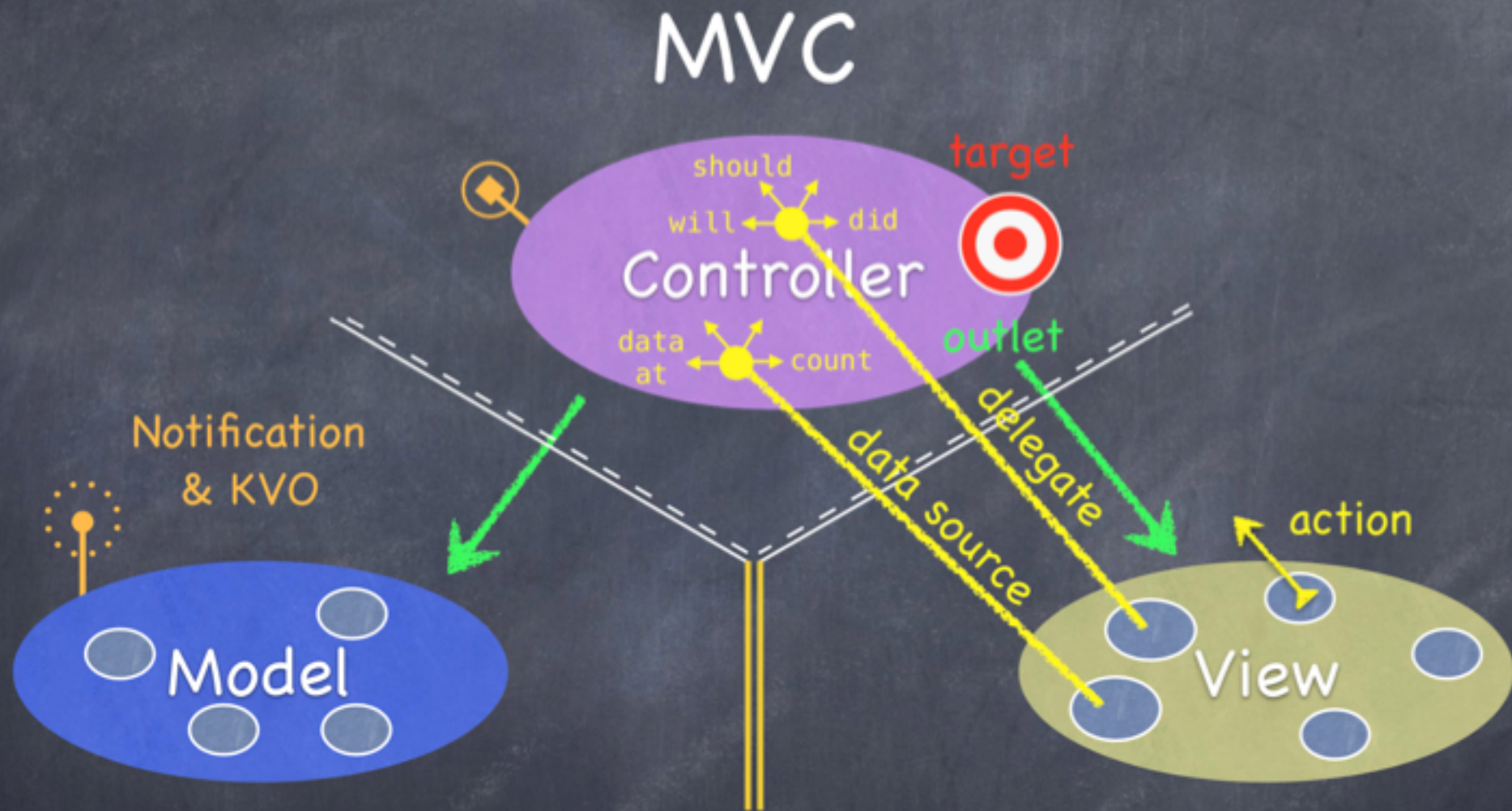
```
let sortedArray = array.sorted(by: >)
```

```
let sortedArray = array.filter({ $0 > 5 }).sorted(by: >).map({ $0 * $0 })
```

# MVC

- Model (模型): App 的逻辑、功能
- Controller (控制器): UI 的逻辑 (模型是如何与用户交互的)
- View (视图): App 的展示 (用户具体看到的)

# MVC in iOS



*Demo*

XY-3241



# 学习资料

- Developing iOS 11 Apps with Swift (Stanford CS193P, iTunes U)
- <https://swift.org>
- The Swift Programming Language (iBooks Store)
- <https://www.raywenderlich.com>