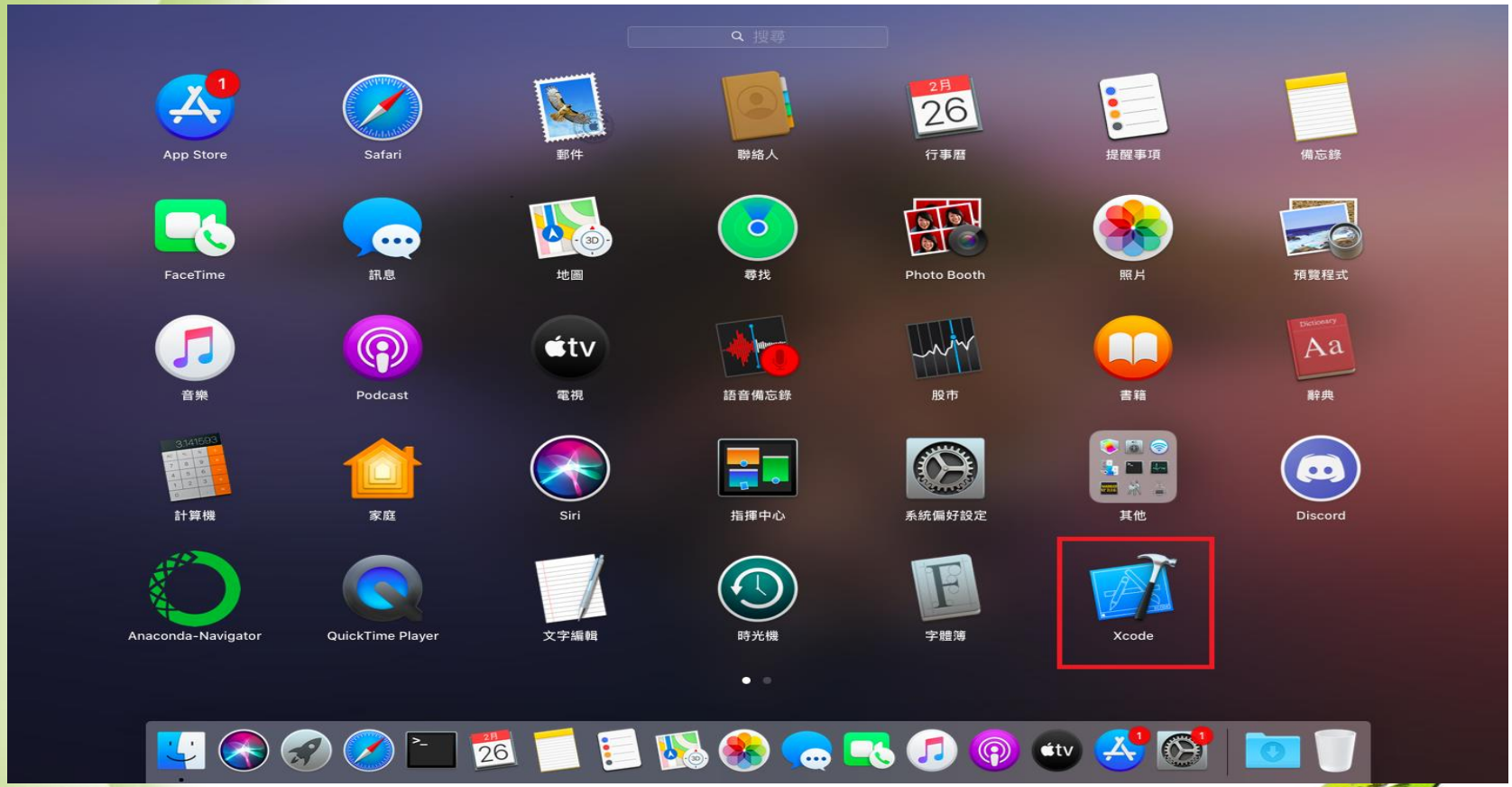
Syntax intro

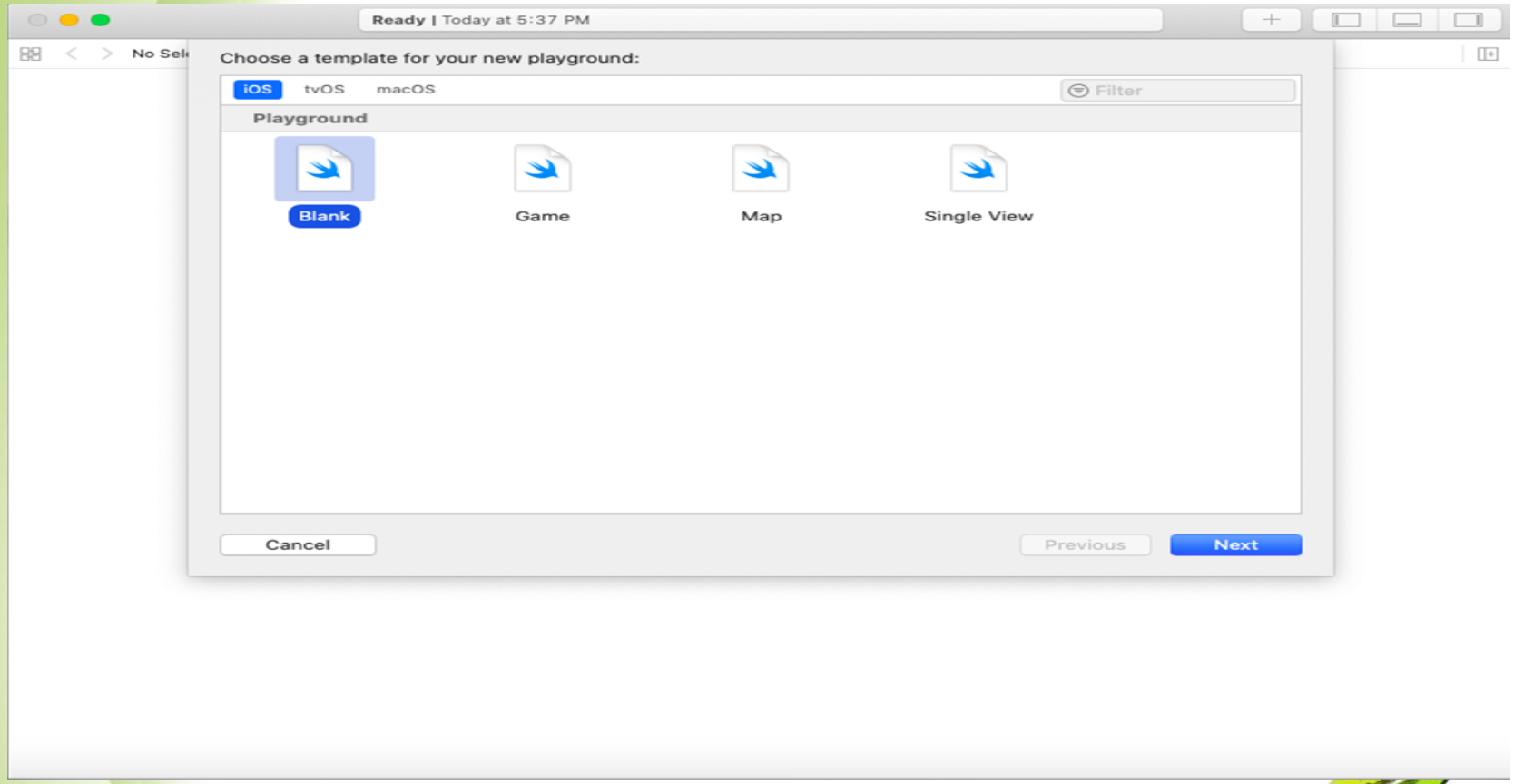
xcode


搜尋
App Store
Safari
郵件
聯絡人
行事曆
提醒事項
備忘錄
FaceTime
訊息
地圖
尋找
Photo Booth
照片
預覽程式
音樂
Podcast
電視
語音備忘錄
股市
書籍
辭典
計算機
家庭
Siri
指揮中心
系統偏好設定
其他
Discord
Anaconda-Navigator
QuickTime Player
文字編輯
時光機
字體簿
Xcode

Playground


Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
2月12日 週六 下午9:20
New
Add Files... ⌥⌘A
Add Packages...
Open... ⌘O
Open Recent
Open Quickly... ⇧⌘O
Close Tab ⌘W
Close Document ^⌘W
Close Editor ^⇧⌘W
Close Window Tab ⇧⌘W
Close Workspace ^⌥⌘W
Save ⌘S
Duplicate... ⇧⌘S
Revert to Saved...
Unlock...
Export...
Show in Finder
Open in Tab ⌥⌘O
Open in New Window
Open with External Editor
Packages
Save As Workspace...
Workspace Settings...
Page Setup... ⇧⌘P
Print... ⌘P
Editor ^⌘T
Editor Below ^⌥⌘T
Window Tab ⌘T
Window ⇧⌘T
File... ⌘N
Target...
Playground... ⌥⇧⌘N
Project... ⇧⌘N
Package... ^⇧⌘N
Workspace... ^⌘N
Group ⌥⌘N
Group from Selection
MyPlayground
~/Desktop
MyPlayground.playground
to Xcode
Version 13.2.1 (13C100)
Create a new Xcode project
Create an app for iPhone, iPad, Mac, Apple Watch, or Apple TV.
Clone an existing project
Start working on something from a Git repository.
Open a project or file
Open an existing project or file on your Mac.

Playground


Ready | Today at 5:37 PM
Choose a template for your new playground:
iOS
tvOS
macOS
Filter
Playground
Blank
Game
Map
Single View
Cancel
Previous
Next

Variable Declaration

```
1  import UIKit
2
3  var str = "Hello, playground"
4
5  let str1 = "hello"
6
7  str = "oh"
8
9  str1 = "oh"
```

Cannot assign to value: 'str1' is a 'let' constant

var 宣告為變數
let 宣告為常數

變數初始化值後能再次變更值，而常數不能

按此執行程式

```
1  import UIKit
2
3
4  var str: String
5
6
7  var str1: String = "Hi"
8
9
10 var x = 1.0, y: Double = 2.0, z = 3.0
11
12
```

變數與常數宣告可加或不加型態，也可以選擇是否要給初值

連續宣告，變數之間使用逗點隔開

Print

```
1  import UIKit
2
3
4  var str: String = "Hi"
5  let str1: String = "Jack"
6
7
8  print ("HI Jack")
9  print ("HI " + str1)
10 print ("\(str) Jack")
11
12
13 var str2 = str + " " + str1
14 print (str2)
15
16
```

假如要print變數或常數可以使用' + '或 '\'

String 能用其他String來串接

Control flow

```swift
import UIKit

var condition: Bool = true
var condition2: Int = 1

if condition {

}else if condition2 {

}else {

}
```

基本的if else 用法，但condition部分只能是Bool型態

```swift
import UIKit

var n = 3
switch n{
case 1:
    break
case 2:
    print("2")
default:
    break
}
```

switch 中每個case中一定要有程式碼，如果不需要執行的內容可以使用break跳過

不像C/C++ 每個case最後不一定要加break也會自己跳出switch

Function

```
var Height = 1.7 //variable
var Weight :Double = 45
let BMI : Double //constant
```

```
func BMICalcuator (height: Double, weight: Double) -> Double {
    let BMI = weight / pow(height, 2)
    return BMI
}
BMI = BMICalcuator(height: Height, weight: Weight)
print ("YourBMI = \(BMI)")
```

函式的回傳型態

呼叫函式時參數要對應宣告時的標籤

Optional

```swift
import UIKit

let possibleNumber = "123"
let convertedNumber = Int(possibleNumber)


let possibleNumber2: String? = nil
let possibleNumber3: String? = "456"


print(type(of: convertedNumber))
print(convertedNumber)
```

```
Optional<Int>
Optional(123)
```

```swift
import UIKit

let possibleNumber = "Hello"
let convertedNumber = Int(possibleNumber)

print(type(of: convertedNumber))
print(convertedNumber)
```

```
Optional<Int>
nil
```

假如字串是數字就能夠正常轉為Int

一般的Int是無法設為nil的因此會被設為Optional此資料型態

假如字串不是數字就會回傳nil

使用type能知道目標變數的型態

也可以自行宣告為Optional資料型態，主要是在一般型態後面加上'?'

Optional就好比一個箱子可能有東西也可能沒有，要打開才能知道

Optional

```
1  import UIKit
2
3  let possibleNumber = "123"
4  let convertedNumber = Int(possibleNumber)
5
6  print(convertedNumber!)
7
8
9
```

我們可以在變數後面加上'!'來將Optional解開取得裡面的資料

```
1  import UIKit
2
3  let possibleNumber = "hello"
4  let convertedNumber = Int(possibleNumber)
5
6  print(convertedNumber!)   error: Execution was i
7
8
9
```

但假如解開後發現是nil在某些地方會出錯

Optional

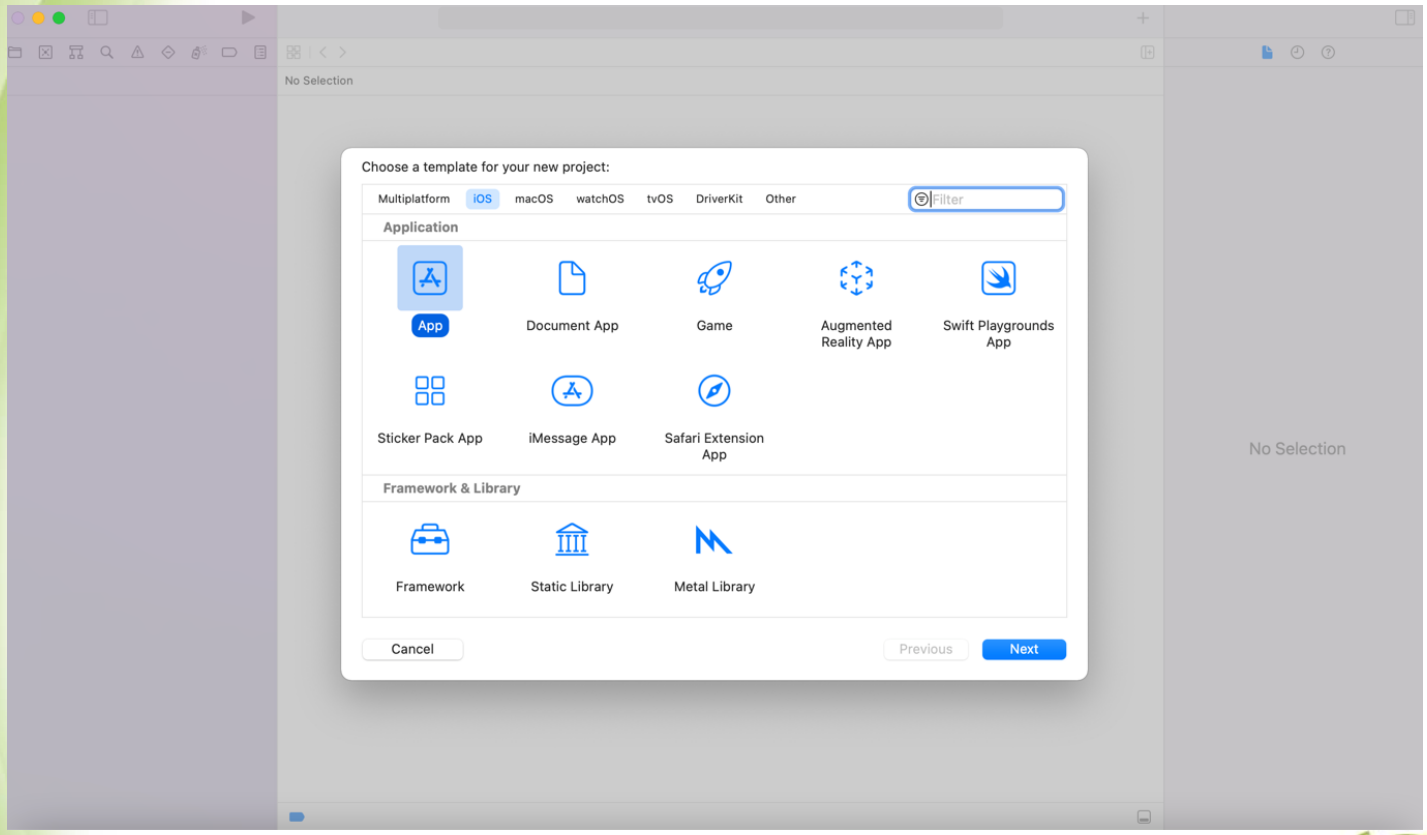
```swift
import UIKit

let possibleNumber = "123"
let convertedNumber = Int(possibleNumber)

if let Number = convertedNumber{
    print(Number)
} else {
    print("無法轉換為數字")
}
```

另一種方式是用if let 生成常數去接收Optional這樣比較能避免出錯

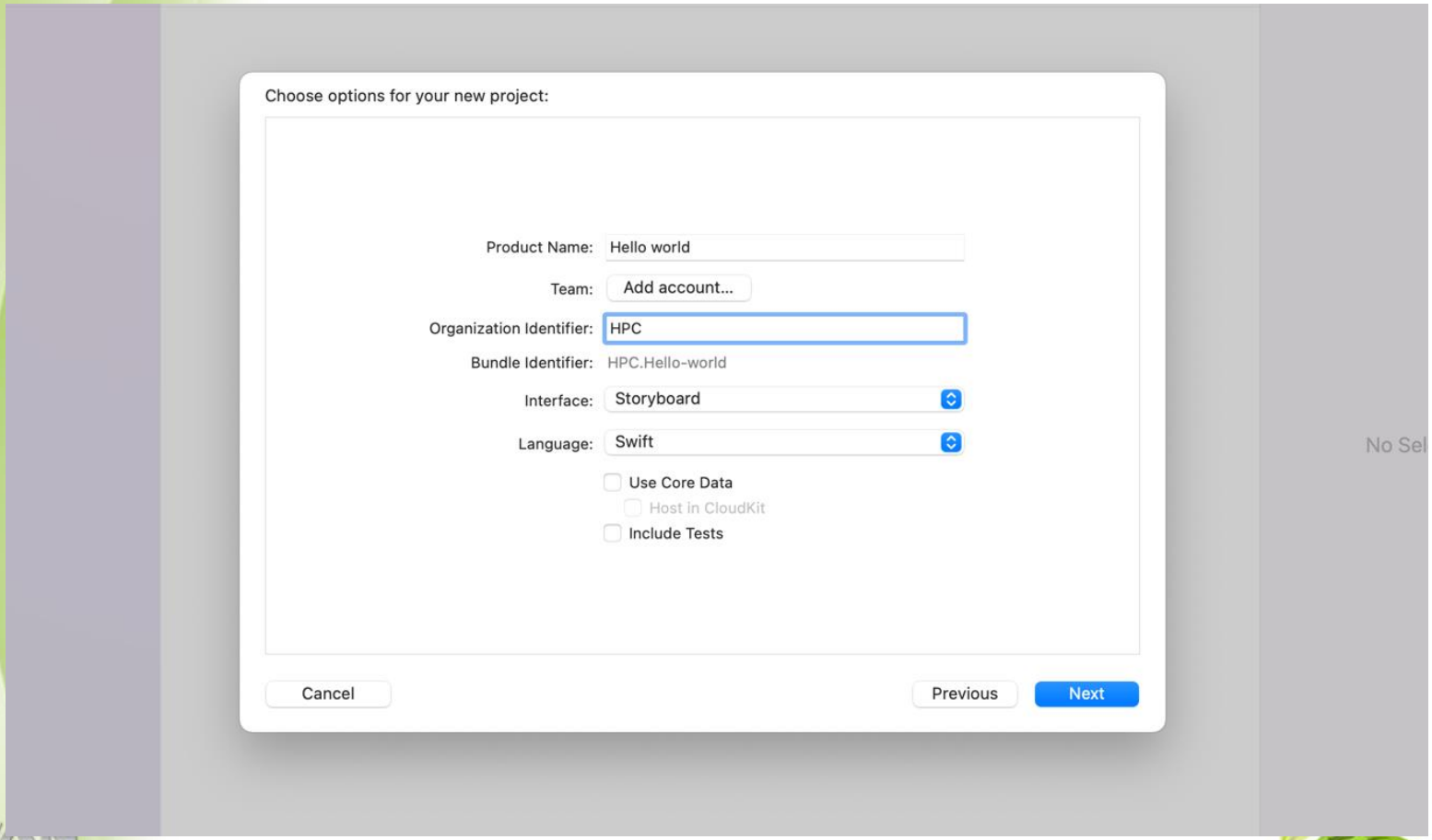
Hello World


Welcome to Xcode
Version 13.2.1 (13C100)
Create a new Xcode project
Create an app for iPhone, iPad, Mac, Apple Watch, or Apple TV.
Clone an existing project
Start working on something from a Git repository.
Open a project or file
Open an existing project or file on your Mac.
Show this window when Xcode launches
Segue
~/Desktop/2018 iOS/class03-1
AlertWithInput
~/Desktop/2018 iOS/class02-2
Alert
~/Desktop/2018 iOS/class02-2
Date Picker
~/Desktop/2018 iOS/class02-2
Hello world
~/Desktop/2018 iOS/class01-1
class02-1
~/Desktop/2018 iOS/class02-1
Open another project...

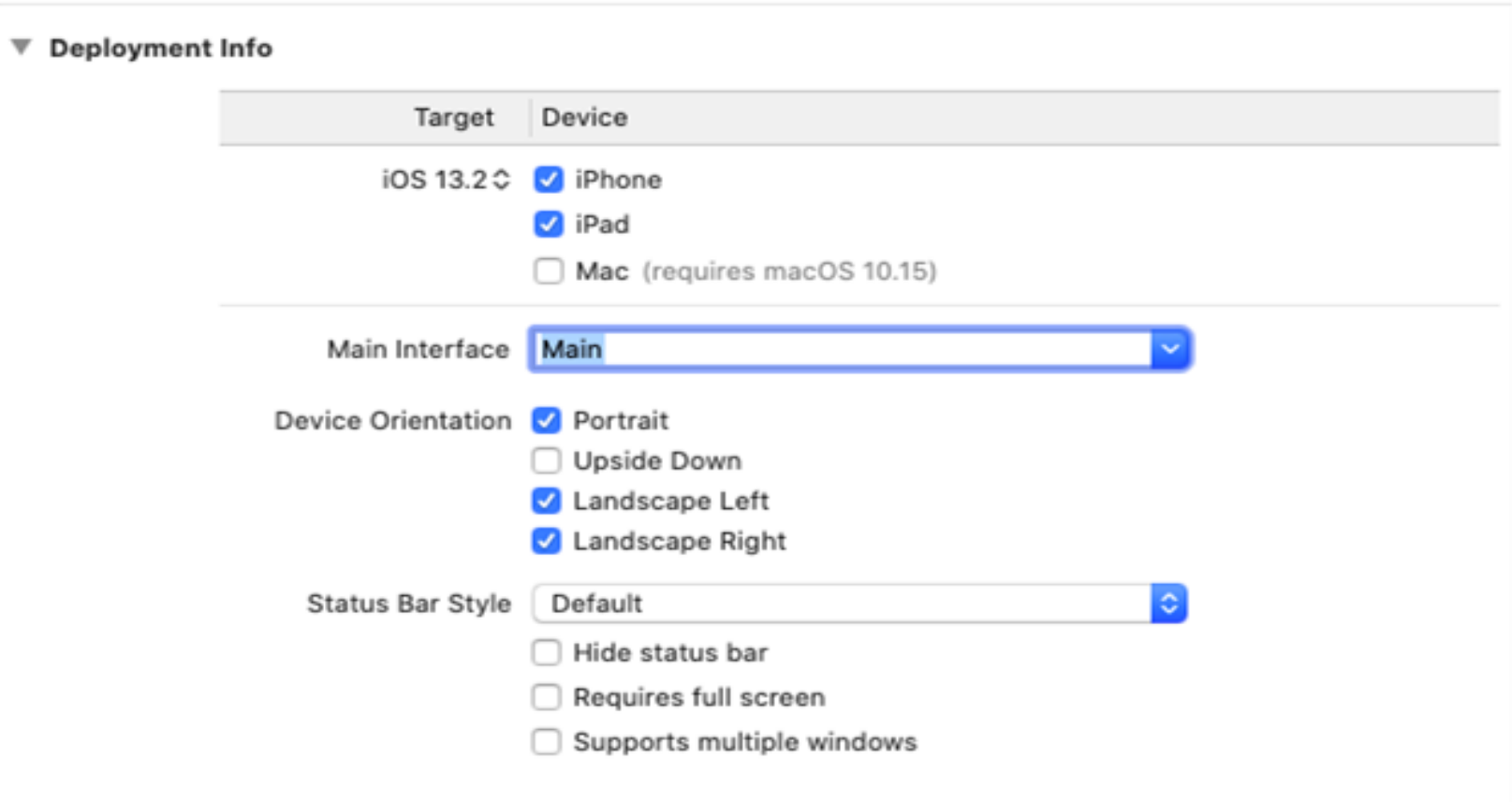
Hello World


No Selection
Choose a template for your new project:
Multiplatform
iOS
macOS
watchOS
tvOS
DriverKit
Other
Filter
Application
App
Document App
Game
Augmented Reality App
Swift Playgrounds App
Sticker Pack App
iMessage App
Safari Extension App
Framework & Library
Framework
Static Library
Metal Library
Cancel
Previous
Next
No Selection

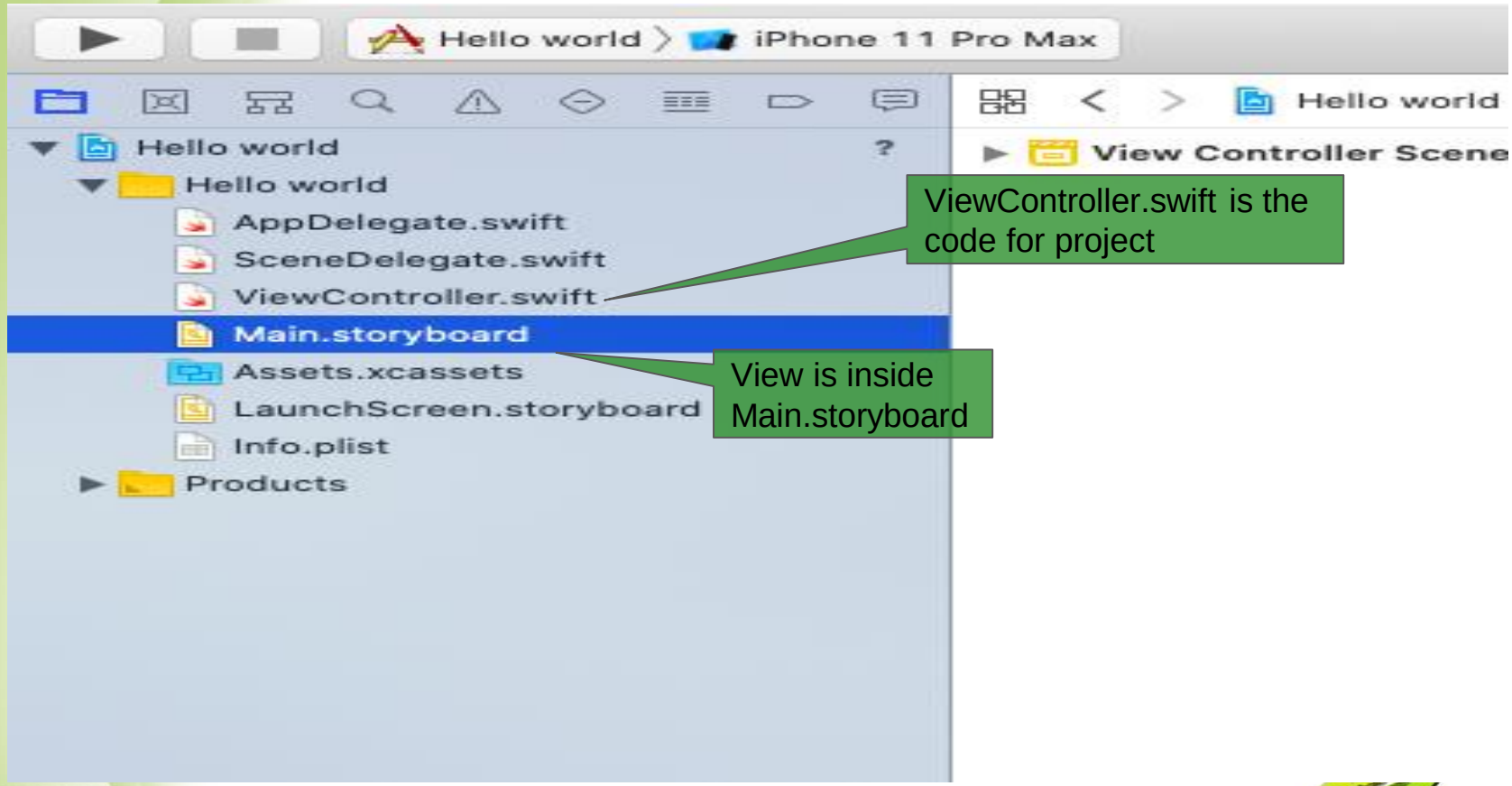
Hello World


Choose options for your new project:
Product Name: Hello world
Team: Add account...
Organization Identifier: HPC
Bundle Identifier: HPC.Hello-world
Interface: Storyboard
Language: Swift
Use Core Data
Host in CloudKit
Include Tests
Cancel
Previous
Next

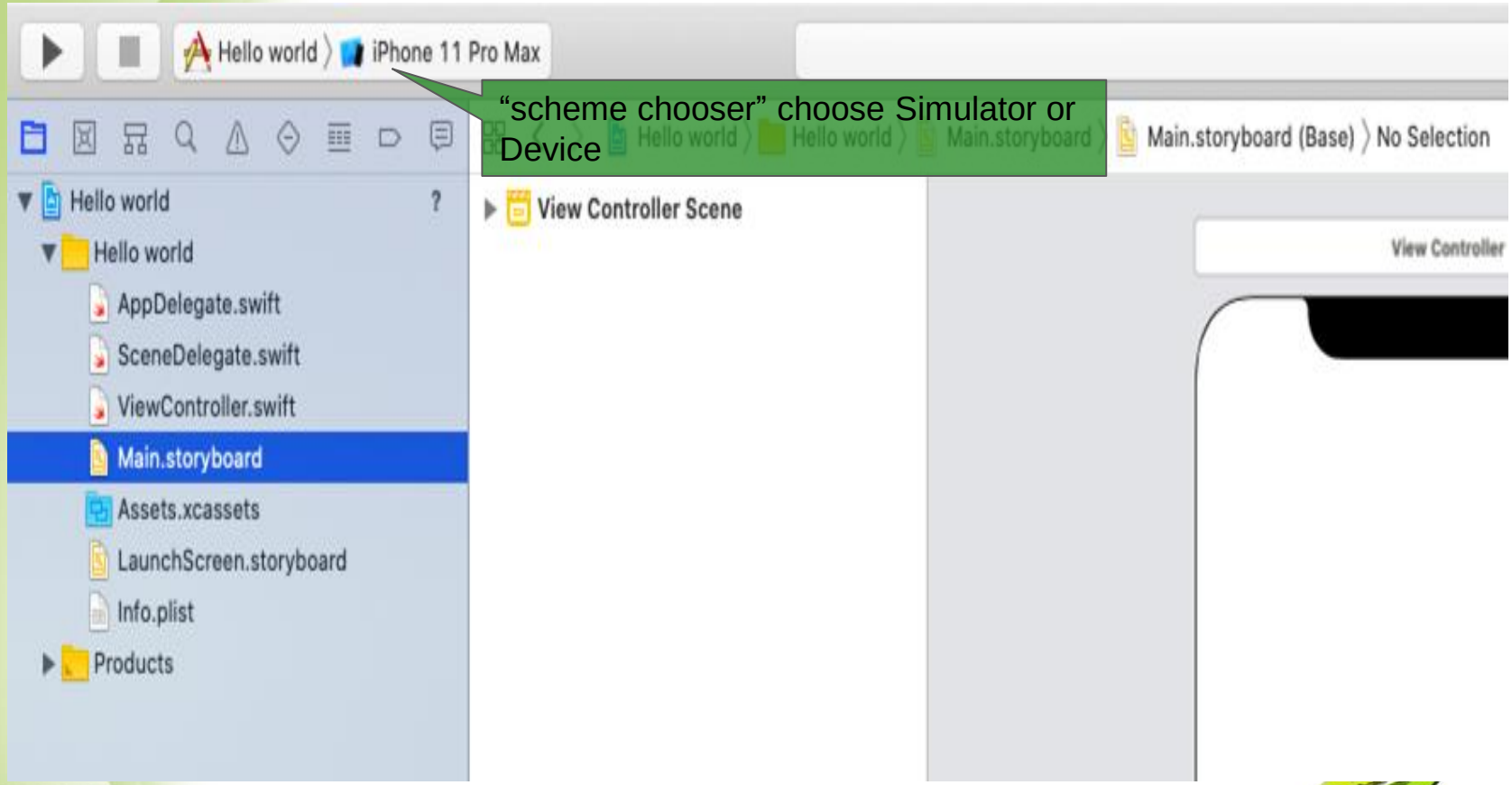
Hello World


Deployment Info
Target
Device
iOS 13.2
iPhone
iPad
Mac (requires macOS 10.15)
Main Interface
Main
Device Orientation
Portrait
Upside Down
Landscape Left
Landscape Right
Status Bar Style
Default
Hide status bar
Requires full screen
Supports multiple windows

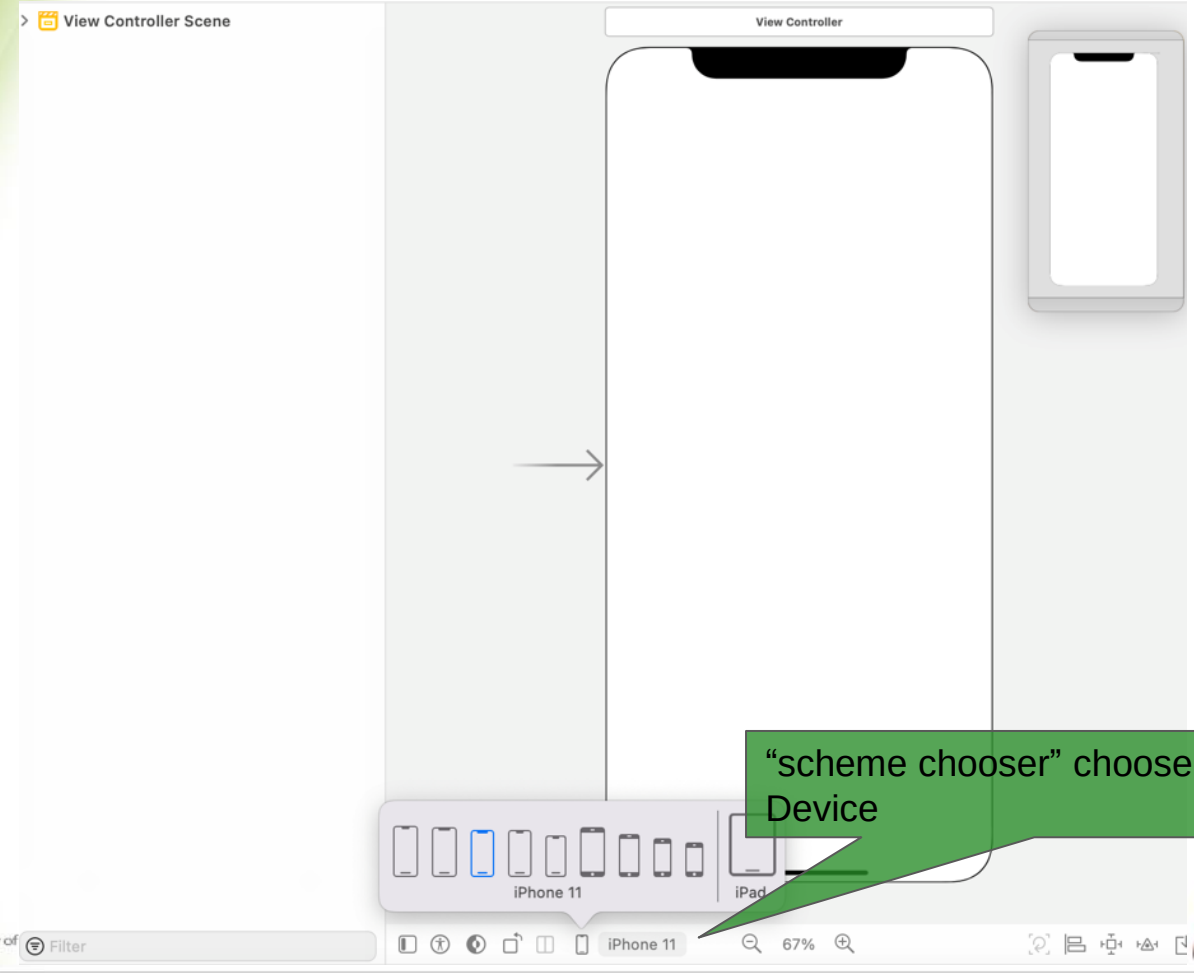
Hello World


Hello world
iPhone 11 Pro Max
Hello world
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
Hello world
View Controller Scene
ViewController.swift is the code for project
View is inside Main.storyboard

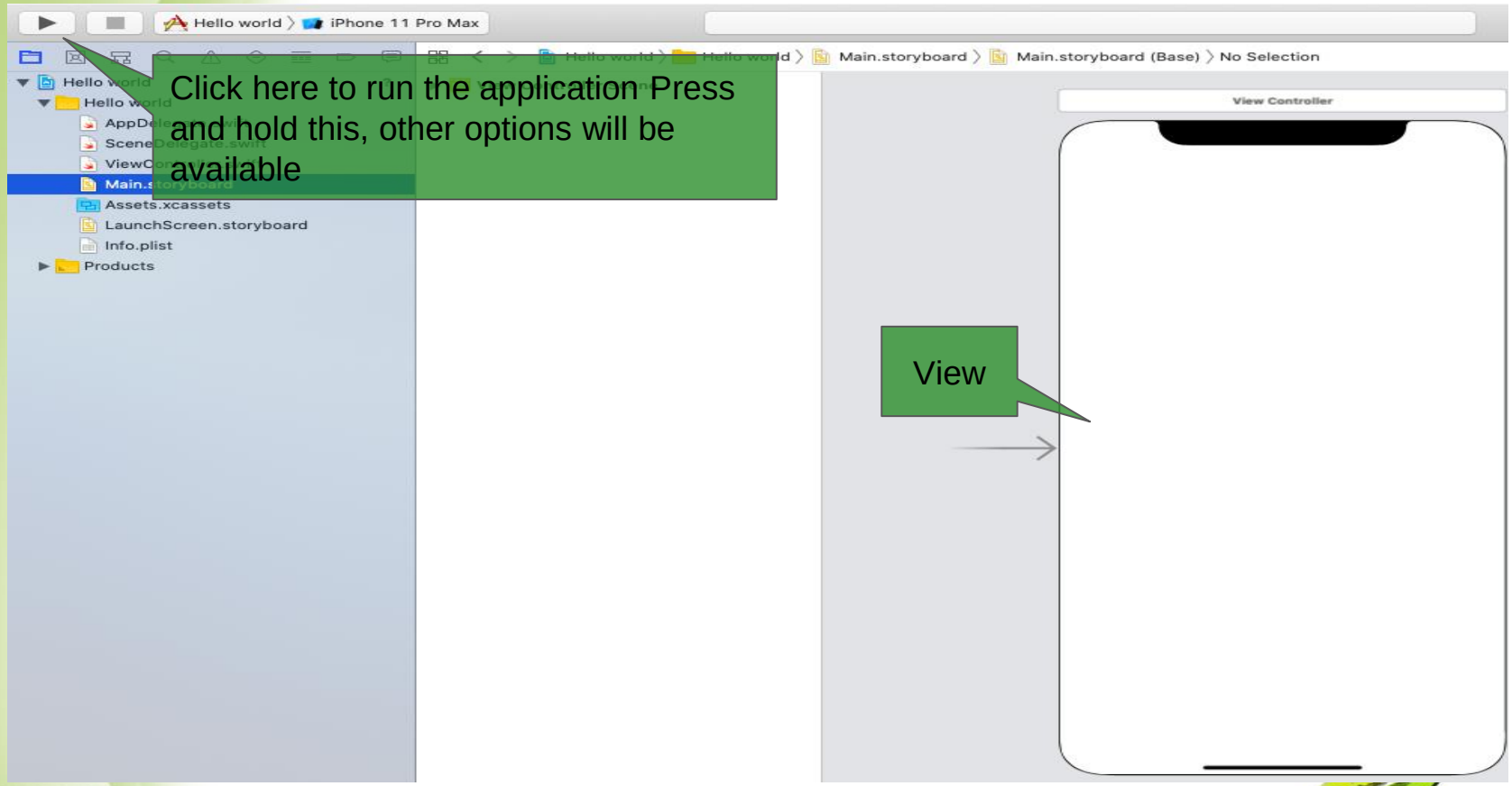
Hello World


Hello world
iPhone 11 Pro Max
“scheme chooser” choose Simulator or Device
Hello world
Hello world
Main.storyboard
Main.storyboard (Base)
No Selection
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
View Controller Scene
View Controller

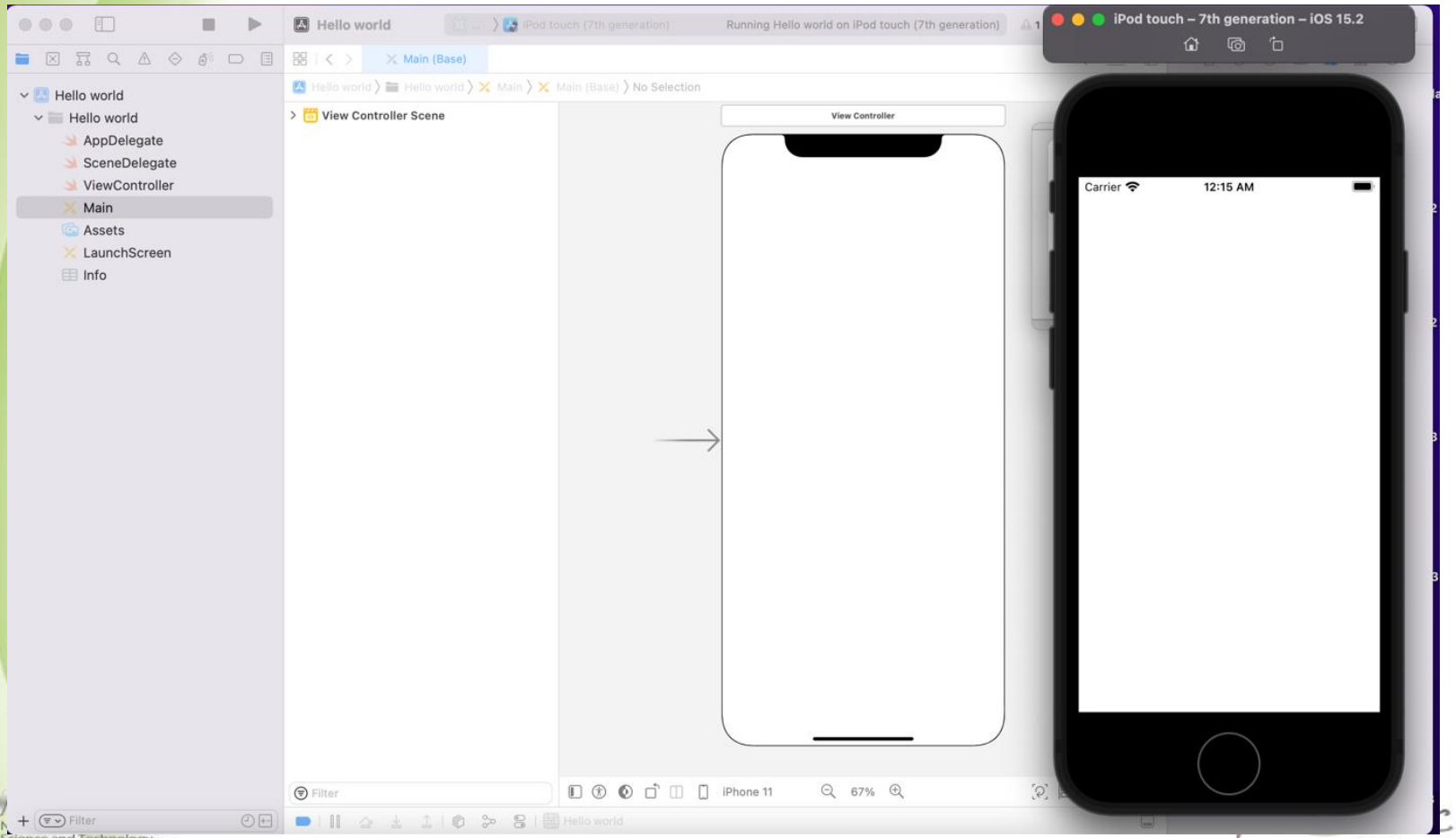
Hello World


View Controller Scene
View Controller
iPhone 11
iPad
iPhone 11
67%
“scheme chooser” choose Simulator or Device

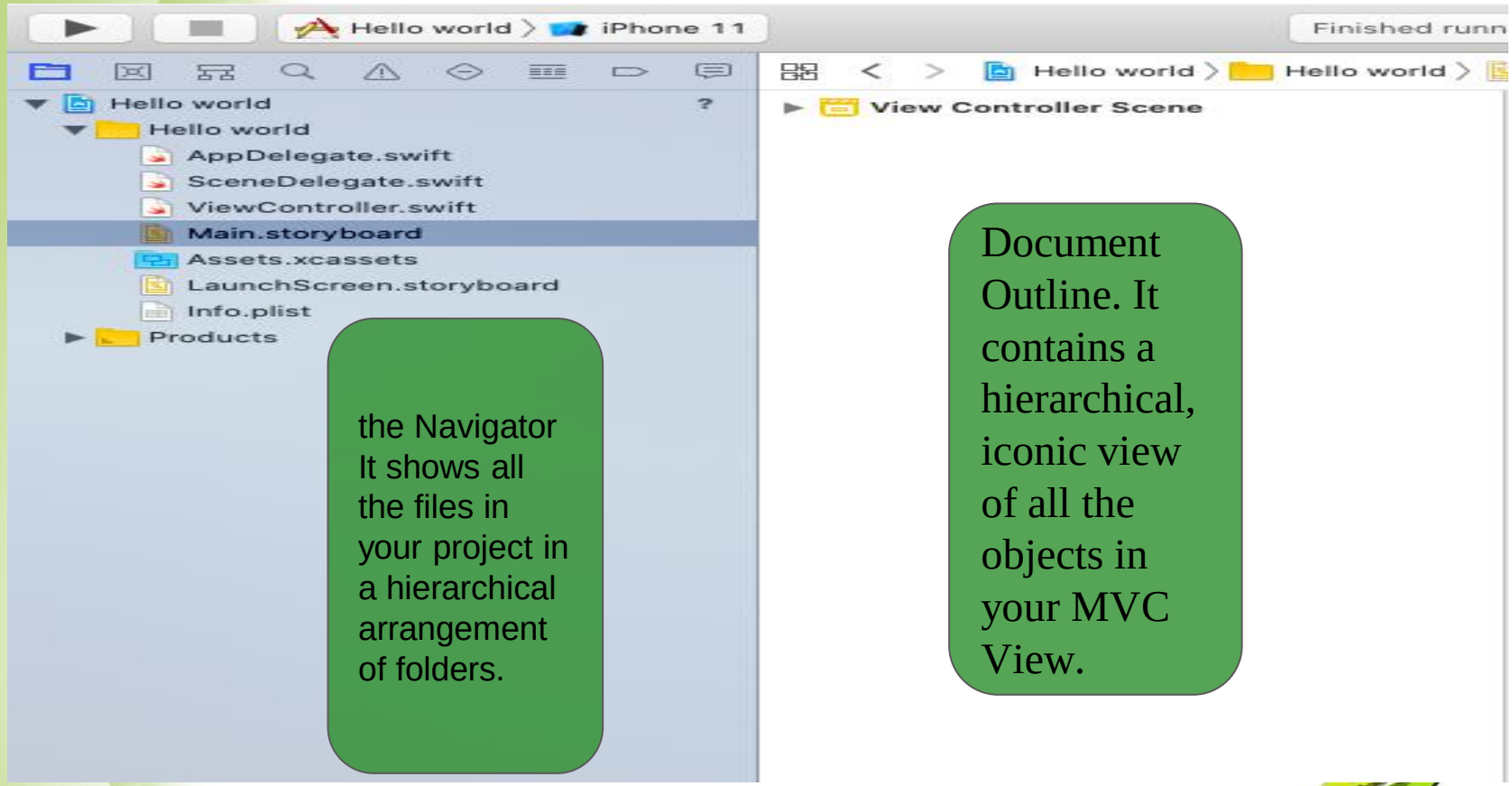
Hello World


Hello world
iPhone 11 Pro Max
Main.storyboard
Main.storyboard (Base)
No Selection
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
Click here to run the application Press and hold this, other options will be available
View Controller
View

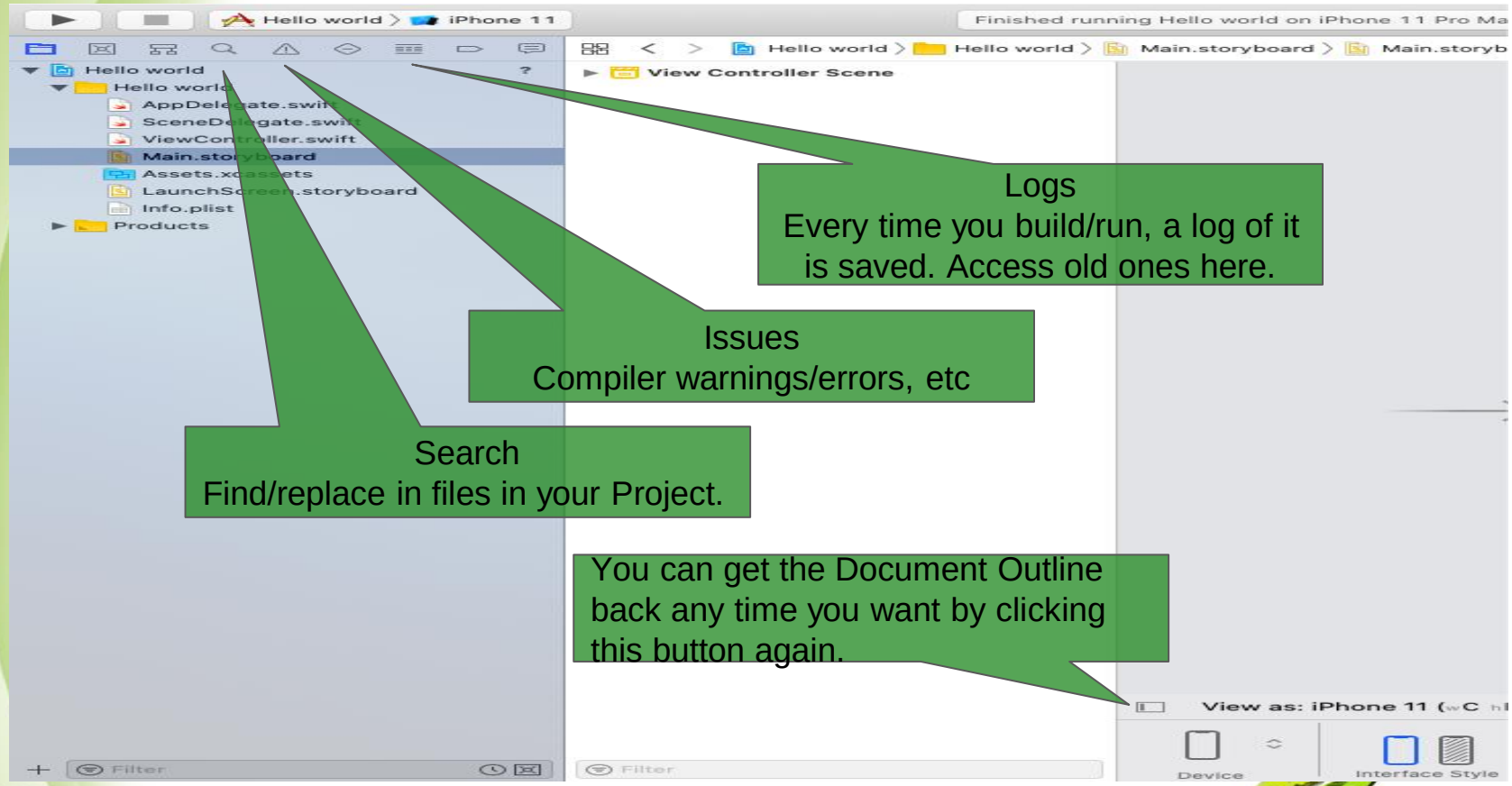
Hello World


Hello world
Running Hello world on iPod touch (7th generation)
Main (Base)
Hello world
Hello world
AppDelegate
SceneDelegate
ViewController
Main
Assets
LaunchScreen
Info
View Controller Scene
No Selection
View Controller
iPhone 11
67%
iPod touch – 7th generation – iOS 15.2
Carrier
12:15 AM

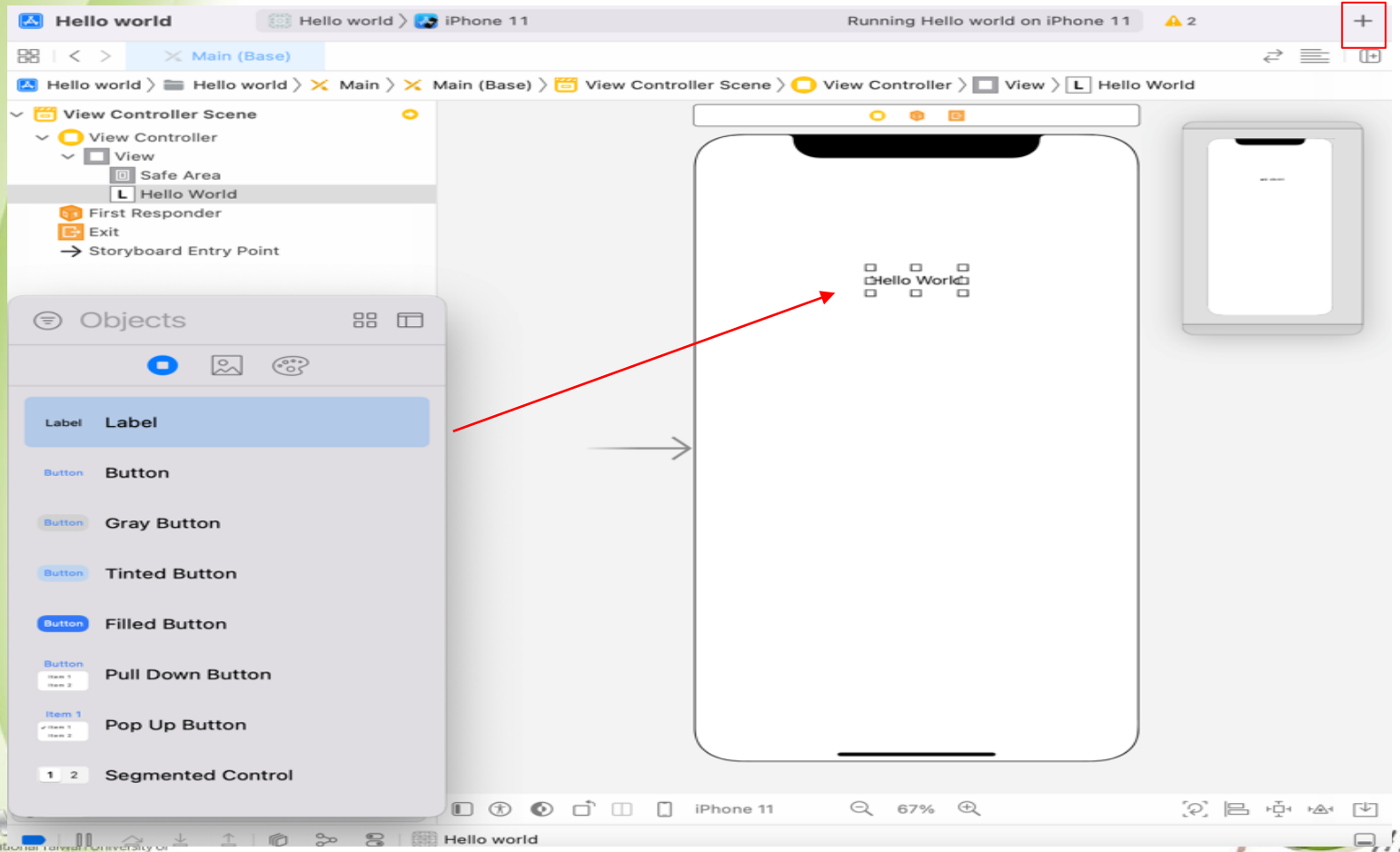
Hello World


Hello world
iPhone 11
Finished runn
Hello world
Hello world
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
View Controller Scene
the Navigator It shows all the files in your project in a hierarchical arrangement of folders.
Document Outline. It contains a hierarchical, iconic view of all the objects in your MVC View.

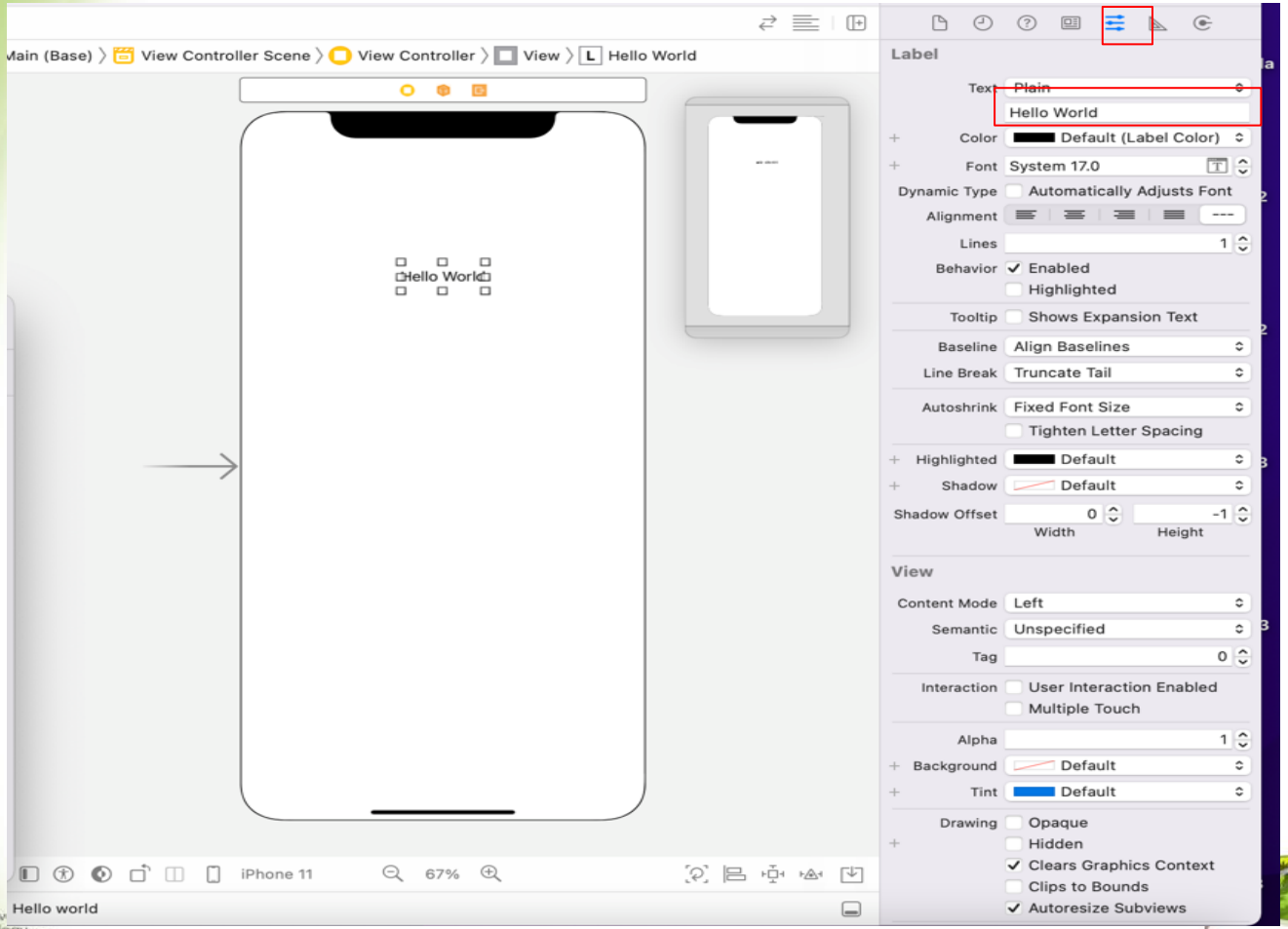
Hello World


Hello world > iPhone 11
Finished running Hello world on iPhone 11 Pro Ma
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
Filter
Hello world > Hello world > Main.storyboard > Main.storyb
View Controller Scene
Filter
View as: iPhone 11
Device
Interface Style
Logs
Every time you build/run, a log of it is saved. Access old ones here.
Issues
Compiler warnings/errors, etc
Search
Find/replace in files in your Project.
You can get the Document Outline back any time you want by clicking this button again.

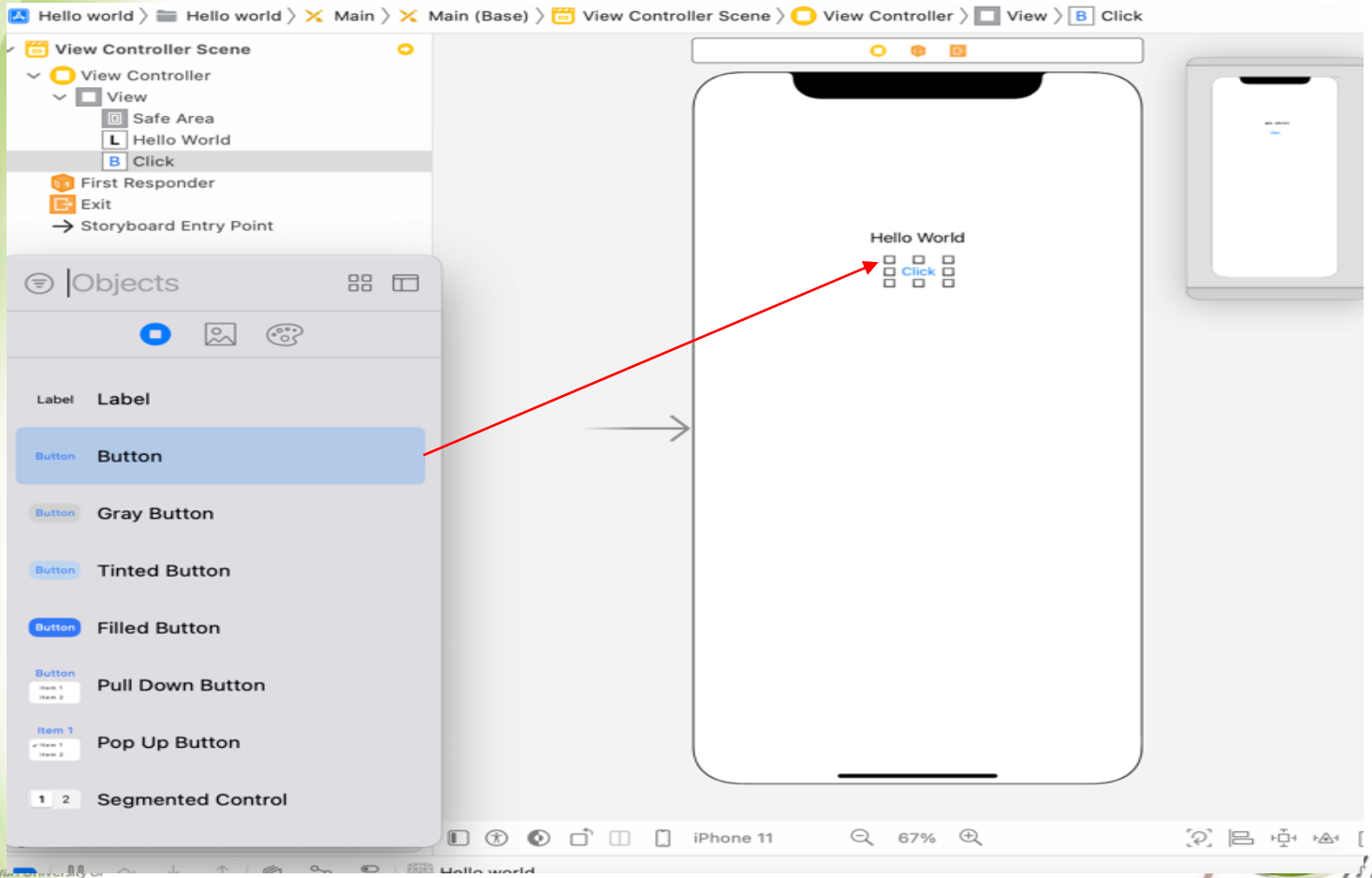
Hello World



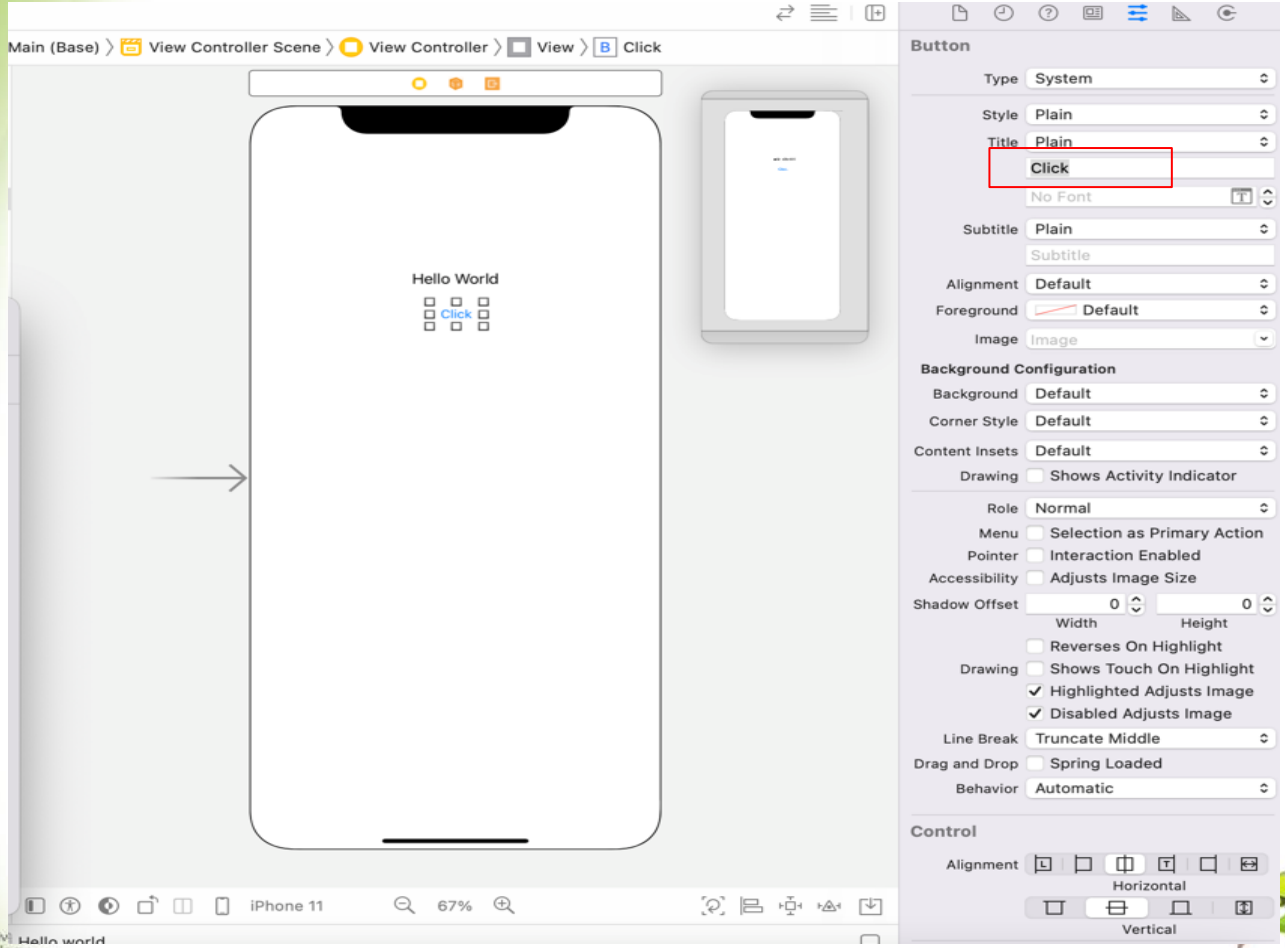
Hello World



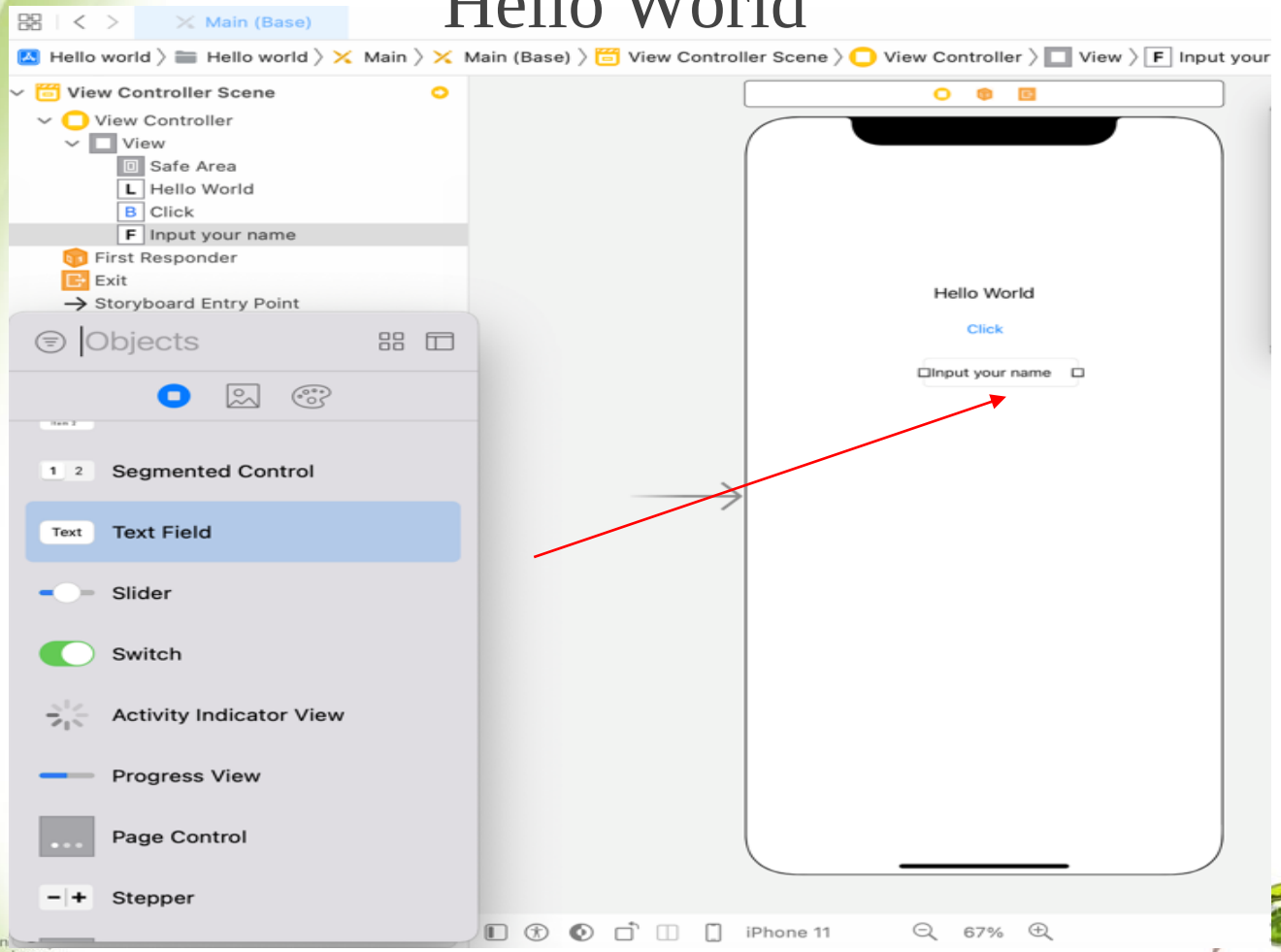
Hello World


Hello world
Hello world
Main
Main (Base)
View Controller Scene
View Controller
View
Click
View Controller Scene
View Controller
View
Safe Area
Hello World
Click
First Responder
Exit
Storyboard Entry Point
Objects
Label
Label
Button
Button
Gray Button
Tinted Button
Filled Button
Pull Down Button
Pop Up Button
Segmented Control
Hello World
Click
iPhone 11
67%

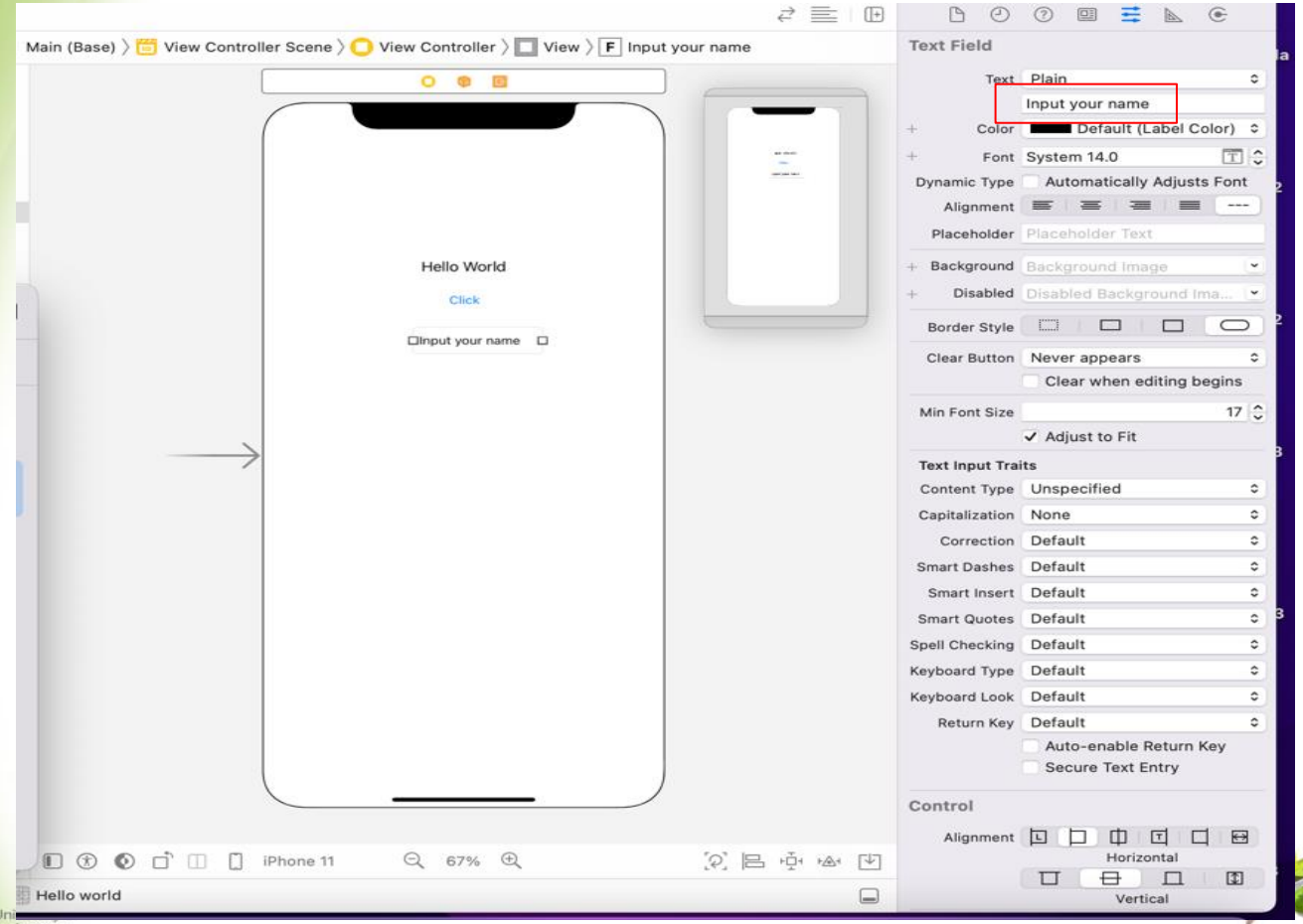
Hello World



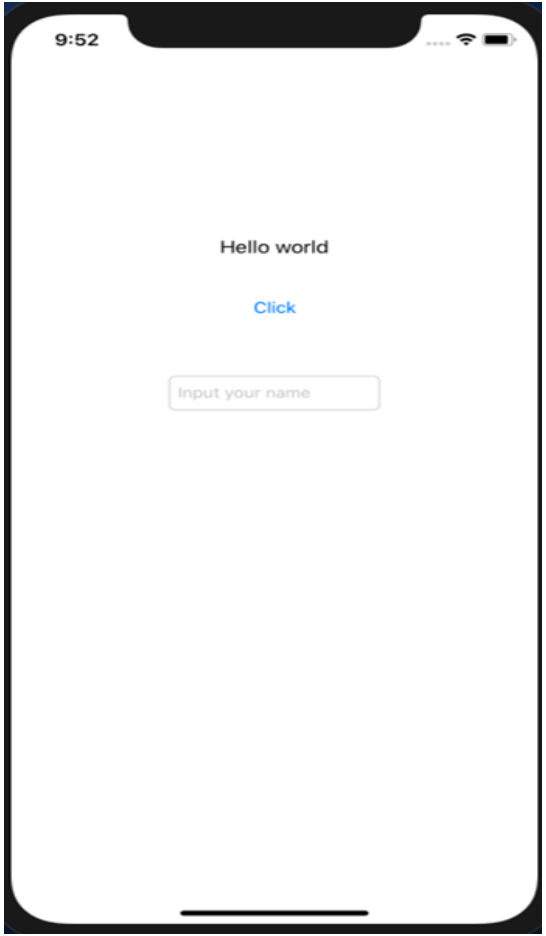
Hello World



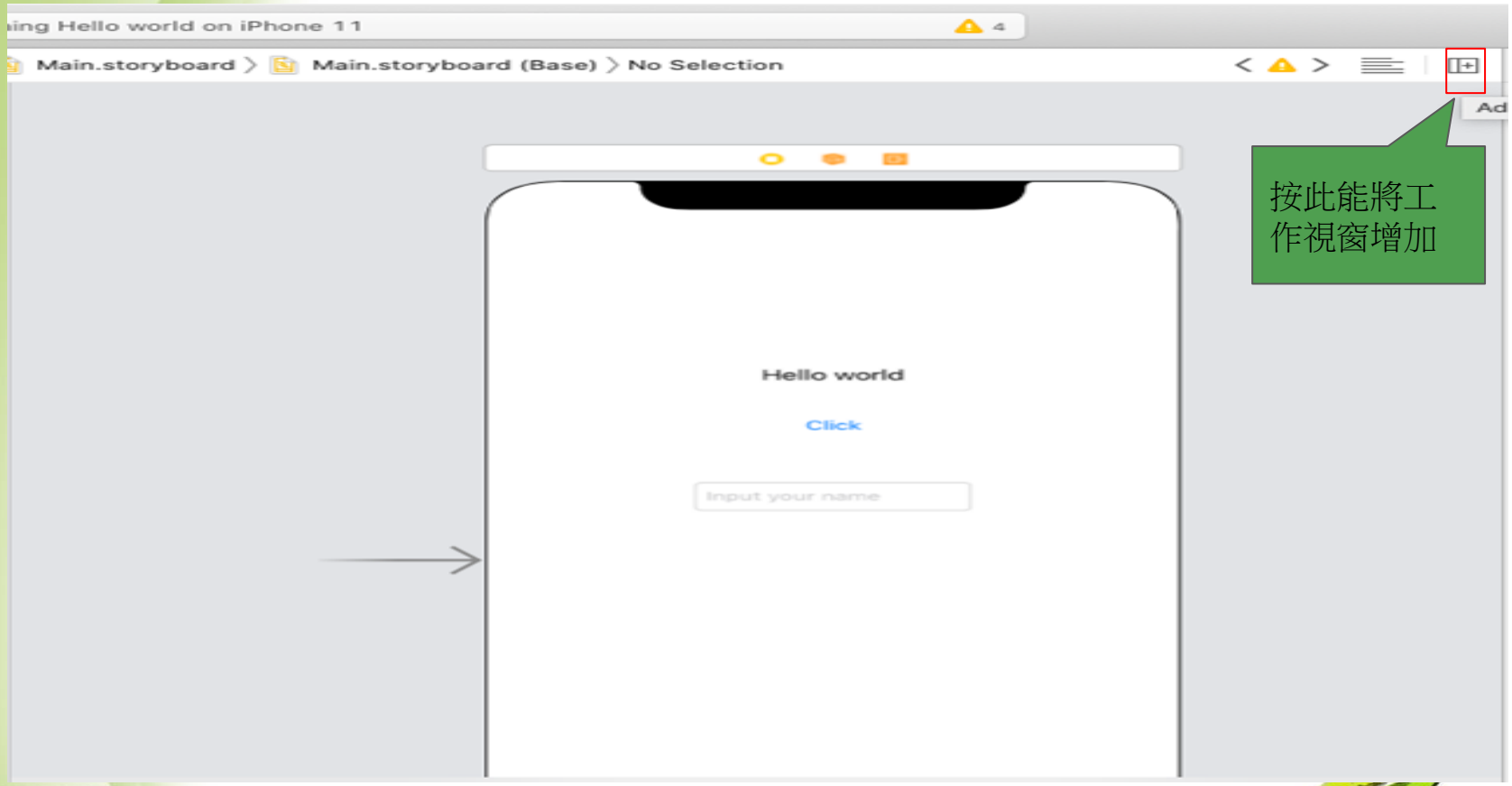
Hello World



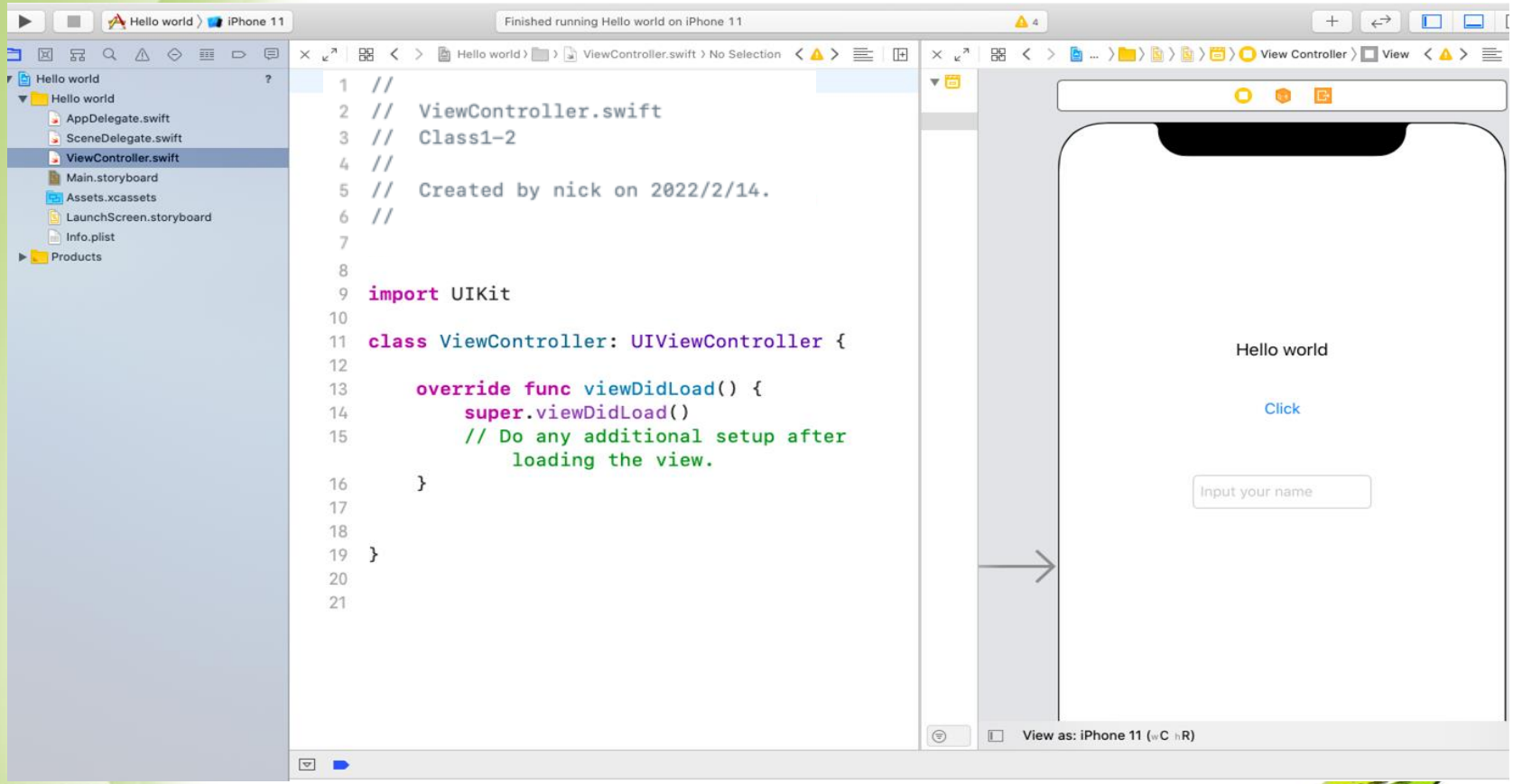
Hello World



Hello World


ing Hello world on iPhone 11
4
Main.storyboard
Main.storyboard (Base)
No Selection
Ad
按此能將工作視窗增加
Hello world
Click
Input your name

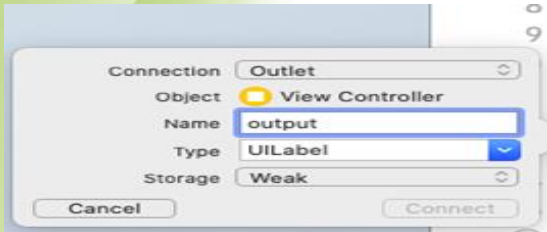
Hello World

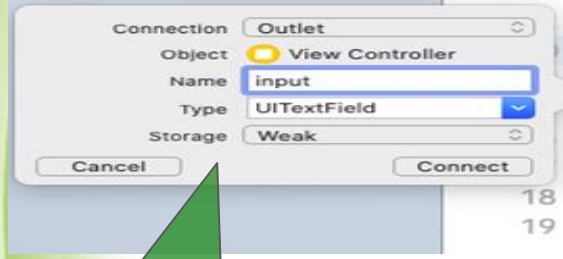

Hello world
iPhone 11
Finished running Hello world on iPhone 11
Hello world
Hello world
AppDelegate.swift
SceneDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
Hello world › ViewController.swift › No Selection
View Controller › View
//
// ViewController.swift
// Class1-2
//
// Created by nick on 2022/2/14.
//
import UIKit
class ViewController: UIViewController {
override func viewDidLoad() {
super.viewDidLoad()
// Do any additional setup after loading the view.
}
}
Hello world
Click
Input your name
View as: iPhone 11 (wC hR)

Hello World


Finished running Hello world on iPhone 11
ViewController.swift
viewDidLoad()
View
Hello world
//
// ViewController.swift
// Class1-2
//
// Created by nick on 2022/2/14.
//
import UIKit
class ViewController: UIViewController {
override func viewDidLoad() {
super.viewDidLoad()
// Do any additional setup after loading the view.
}
}
Hello world
Click
點選並按住control拖移至左側
View as: iPhone 11

Hello World






```
import UIKit

class ViewController: UIViewController {

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

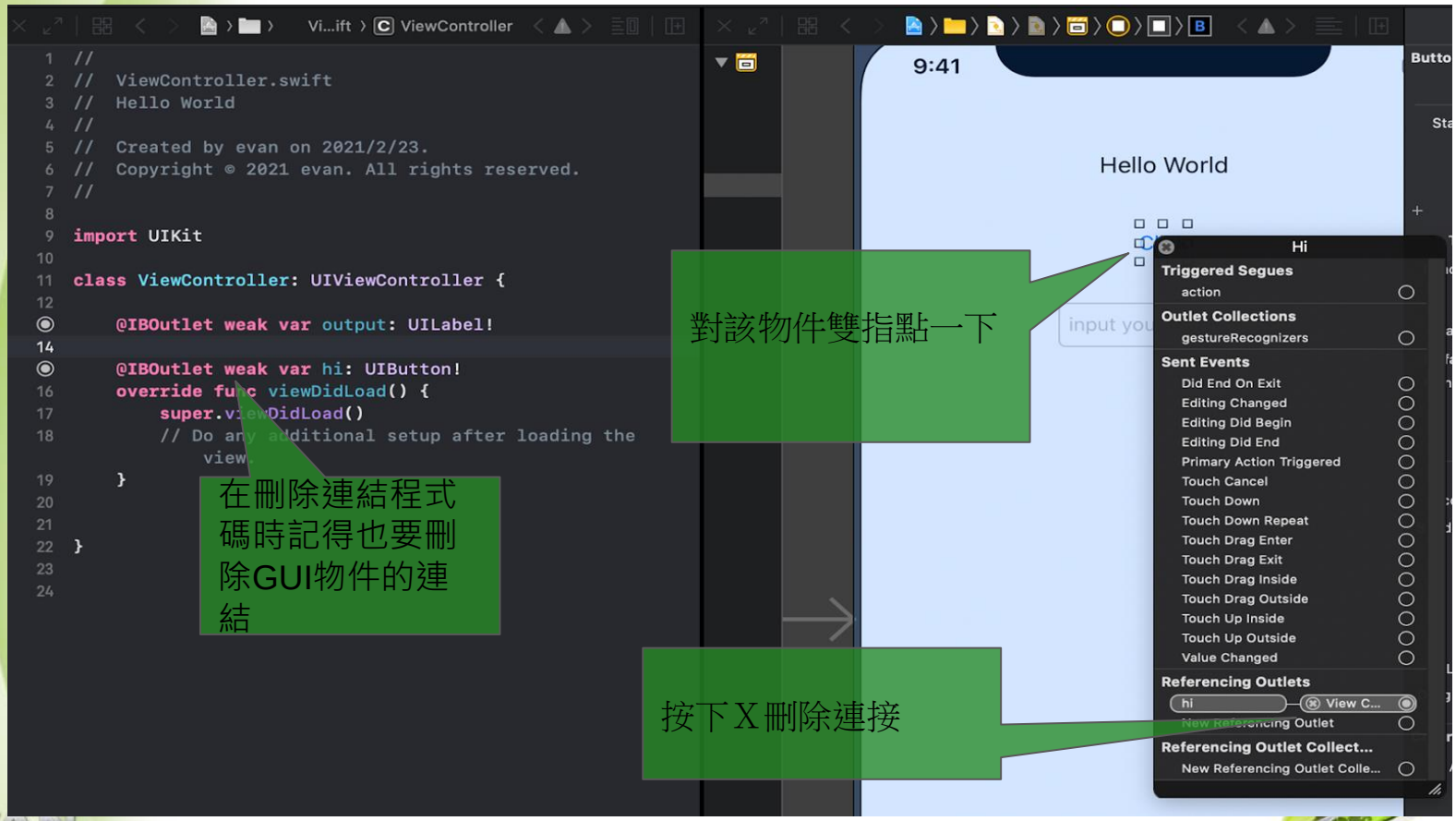
```
class ViewController: UIViewController {

    @IBOutlet weak var output: UILabel!

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

將label與TextField連結到viewcontroller

```
class ViewController: UIViewController {

    @IBOutlet weak var output: UILabel!
    @IBOutlet weak var input: UITextField!

    override func viewDidLoad() {
```

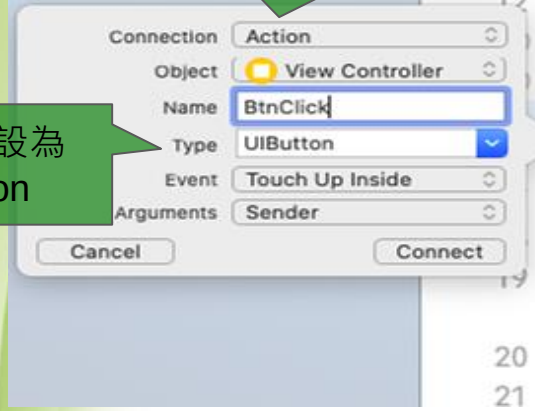
連結後會產生初這兩條宣告

Hello World


ViewController
ViewController.swift
Hello World
Created by evan on 2021/2/23.
Copyright © 2021 evan. All rights reserved.
import UIKit
class ViewController: UIViewController {
@IBOutlet weak var output: UILabel!
@IBOutlet weak var hi: UIButton!
override func viewDidLoad() {
super.viewDidLoad()
// Do any additional setup after loading the view
}
}
9:41
Hello World
Hi
Triggered Segues
action
Outlet Collections
gestureRecognizers
Sent Events
Did End On Exit
Editing Changed
Editing Did Begin
Editing Did End
Primary Action Triggered
Touch Cancel
Touch Down
Touch Down Repeat
Touch Drag Enter
Touch Drag Exit
Touch Drag Inside
Touch Drag Outside
Touch Up Inside
Touch Up Outside
Value Changed
Referencing Outlets
hi
View C...
New Referencing Outlet
Referencing Outlet Collect...
New Referencing Outlet Colle...
對該物件雙指點一下
在刪除連結程式碼時記得也要刪除GUI物件的連結
按下X刪除連接

Hello World

button連結是Action



將type設為UIButton

```
11  class ViewController: UIViewController {
12
        @IBOutlet weak var output: UILabel!
        @IBOutlet weak var input: UITextField!


        override func viewDidLoad() {
            super.viewDidLoad()
            // Do any additional setup after
                loading the view.
20      }
21
```

```
        @IBAction func BtnClick(_ sender: UIButton)
            {
17      }
```

產生出Click fuc

Hello World

```
8
9   import UIKit
10
11  class ViewController: UIViewController {
12
13      @IBOutlet weak var output: UILabel!
14      @IBOutlet weak var input: UITextField!
15      @IBAction func BtnClick(_ sender: UIButton) {
16          output.text = "Hello, \(input.text)"
17      }
18      override func viewDidLoad() {
19          super.viewDidLoad()
20          // Do any additional setup after loading the
                view.
21      }
22
23
24  }
25
26
```

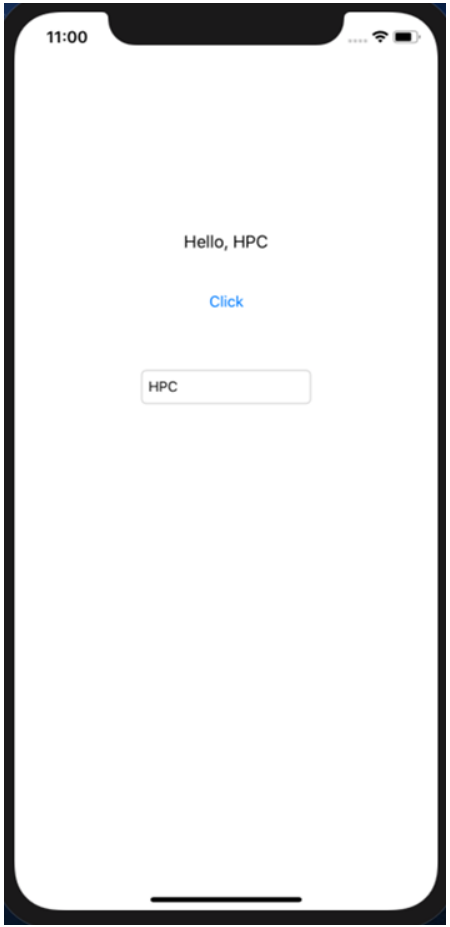
將labal的顯示
"Hello, TextField的文字"

此處的
input.text為
Optional因此
要將他拆開

Hello World

```
 9  import UIKit
10
11  class ViewController: UIViewController {
12
        @IBOutlet weak var output: UILabel!
        @IBOutlet weak var input: UITextField!
        @IBAction func BtnClick(_ sender: UIButton) {
16          output.text = "Hello, \(input.text!)"
17      }
18      override func viewDidLoad() {
19          super.viewDidLoad()
20          // Do any additional setup after loading the
                view.
21      }
```

Hello World



Hello World

```
8
9   import UIKit
10
11  class ViewController: UIViewController {
12
        @IBOutlet weak var output: UILabel!
        @IBOutlet weak var input: UITextField!
        @IBAction func BtnClick(_ sender: UIButton) {
            let controller = UIAlertController(title: "您還未輸入名字！",
                message: "請輸入名字", preferredStyle: .alert)
            let okAction = UIAlertAction(title: "OK",style: .default,handler:
                nil)
            controller.addAction(okAction)

            if input.text! != "" {
                output.text = "Hello,\(input.text!)"
            } else {
                present(controller, animated: true, completion: nil)
            }
        }
```

先宣告一個UIAlert，title決定訊息標題message決定訊息內容

做出一個Alert的互動按鈕

將訊息顯示在畫面

Hello World


11:50
Hello,HPC
Click
Input your name
您還未輸入名字!
請輸入名字
OK
iPhone 11 — 13.3