

swift UI requires new version Mac OS
 september → release version July → beta version
 shift + command + L → objects window
 select view → use recommended constraints
 control + drag → link UI and code
 ex. imageView → rename will break the code, disconnect the outlet first
 ⚡️ IBOutlet weak var imageView: UIImageView!
 override func viewDidLoad() {
 super.viewDidLoad()
 }
 ⚡️ IBAction weak var buttonClicked(_ sender: Any) {
 imageView.image = UIImage(named: "metallica2")
 }
 ⚡️ UIImage ← object

// Array, collection of data types
 var myFavMovie = ["Pulp fiction", "Kill Bill"]
 var myFavMovie = ["Pulp fiction", "Kill Bill", 5] as [Any]
 myFavMovie[1] // "Kill Bill"
 // empty array → var myArray = [String]()
 * when use array will not be able reach some methods
 myFavMovie.count
 .last
 .sort()

// Set ← kind of array but unique values, unordered collections, can not have duplicated values
 var mySet: Set = [1, 2, 3, 4, 5]
 var myInternetSet = Set(myInternetArray) // convert array to set
 var mySet3 = mySet1.union(mySet2) // combine two sets together
 // empty set → var empty = Set<Int>()

// Dictionaries key-value pairing
 // empty dictionaries → var empty = [String: Int]()
 var myFavoriteDirectors = ["1": "A", "2": "B"]

myFavoriteDirectors["1"] // "A"

// while loops
 import UIKit
 var myNumber = 1
 myNumber += 1 // my number is 2

// for Loop, usually works with arrays
 var myFruitArray = ["Banana", "apple", "orange"]
 for fruit in myFruitArray {
 print(fruit)
 }

// if controls <, >, ==, !=, ||, &&
 var myAge = 32
 if myAge < 30 {
 print("so young")
 } else if myAge > 30 & & < 40 {
 print("30s")
 } else {
 print("40+")
 }

// Hold command + click → documentation

// snake_case
 var user_name =
 // CamelCase
 var userName =

// shift + enter → run playground

import UIKit
 var str = "Hi"
 str.append("o")
 str.lowercased()
 var num = 5*4
 print(num)

let userAge = 50 // constant

* can not change the type of the variable after assignment
 * can not use int * double
 * $\frac{\text{int}}{\text{int}} = \text{int}$
 * lossing accuracy
 class name, start with uppercase

let myString: String = "good"
 let anotherNumber: Int = 50

// define
 let myVariable: String

// initialization:
 myVariable = "Test"
 let myUppercaseVariable = myVariable.uppercased()
 print(myUppercaseVariable) // TEST
 print(myVariable) // Test

// similar structure to an array
 var number = 0
 // will run 10 times
 while number < 10 {
 print(number)
 number += 1
 }

// will print 1 to 5,
 for myNewInteger in 1...5 {
 print(myNewInteger)
 }

// functions, coding block

func myFunction() {
 print("my Function")
 }

// input, output, return
 func sumFunction(x: Int, y: Int) -> Int {
 print(x+y) // output
 return x+y // return
 }

// second screen

Shift + command + L → Add view controller

// connect viewController to file

File → New → File... → Cocoa Touch Class → UIViewController

class: SecondViewController

Select the viewController → Identity Inspector → class → SecondViewController
(use the yellow dot)

// How to reach second view from first view

The arrow determines the entry point

Create a button → Control + Drag to second view

// How to create a back button

Choose viewController → Editor → Embed In → Navigation Controller

// View Controller Life Cycle

override func viewDidLoad() {

super.viewDidLoad()

}

override func viewWillAppear(_ animated: Bool) {
 print("")
}

}

viewDidLoad // viewWillAppear only gets called once

viewWillAppear

viewDidAppear

viewWillDisappear

viewDidDisappear

viewWillappear ↴ if go back

viewWillappear

viewDidAppear

// Gesture Recognizers

var isJames = true // outside the func changePic()

override func viewDidLoad() {

super.viewDidLoad()

imageView.isUserInteractionEnabled = true

let gestureRecognizer = UITapGestureRecognizer(target: self,
action: #selector(changePic))

imageView.addGestureRecognizer(gestureRecognizer)

}

@objc func changePic() {

if isJames == true {

imageView.image = UIImage(named: "lars")

myLabel.text = "Lars Ulrich"

isJames = false

} else {

imageView.image = UIImage(named: "james")

myLabel.text = "James Hetfield"

isJames = true

}

// Display Info on the second screen

Use command + B to build the program

text-field + Button

view → yellow button + control + drag

Select segue → Attribute Identifier → rename

var userName = ""

@IBAction func nextClicked(_ sender: Any) {

userName = nameText.text!

performSegue(withIdentifier: "toSecond VC", sender: nil)

}

override func prepare(for segue: UIStoryboardSegue, sender: Any?) {

if segue.identifier == "toSecond VC" {

let destinationViewController = segue.destination as! SecondViewController

destinationVC.myName = userName

}

}

// Alert messages

@IBAction func signUp(_ sender: Any) {

let alert = UIAlertController(title: "Error!", message: "Username!", preferredStyle: UIAlertController.Style.alert)

let okButton = UIAlertAction(title: "OK", style: UIAlertAction.Style.default) { (UIAlertAction) in

print("Button Clicked")

}

alert.addAction(okButton)

self.present(alert, animated: true, completion: nil)

func makeAlert(titleInput: String, messageInput: String) {

// Timers

import UIKit

class ViewController: UIViewController {

@IBOutlet weak var timeLabel: UILabel!

var timer = Timer()

var counter = 0

override func viewDidLoad() {

super.viewDidLoad()

counter = 10

timeLabel.text = "Time: \(counter)"

timer = Timer.scheduledTimer(timeInterval: 1, target: self,
selector: #selector(timerFunction), userInfo: nil, repeats: true)

@objc func timerFunction() {

timeLabel.text = "Time: \(counter)"

counter -= 1

if counter == 0 {

timer.invalidate()

timeLabel.text = "Time's Over!"

@IBAction func buttonClicked(_ sender: Any) {

print("button clicked")