

Piscine Swift - Day 00

Stay Hydrated

Michael BRAVE mbrave@student.42.us.org

42 Staff pedago@42.fr

Summary: This document contains the subject for Day 00 for the "Piscine Swift" from 42

Contents

I	Foreword
II	General Instructions
III	Introduction
IV	Exercise 00: You Are Awesome (“Hello World”)
V	Exercise 01: Buttons That Do Things
VI	Exercise 02: Join The OO Party
VII	Exercise 03: Dates & Times
VIII	Exercise 04: Stay Hydrated App
XI	Bonus: Adding Animations & Graphics

Chapter I

Foreword

According to Wikipedia on Drinking

A daily intake of water is required for the normal physiological functioning of the human body. The USDA recommends a daily intake of total water: not necessarily by drinking but by consumption of water contained in other beverages and foods. The recommended intake is 3.7 liters (appx. 1 gallon) per day for an adult male, and 2.7 liters (appx. 0.75 gallon) for an adult female. Other sources, however, claim that a high intake of fresh drinking water, separate and distinct from other sources of moisture, is necessary for good health – eight servings per day of eight fluid ounces (1.8 liters, or 0.5 gallon) is the amount recommended by many nutritionists, although there is no scientific evidence supporting this recommendation.

Chapter II

General Instructions

- Only this document will serve as reference. Do not trust rumors.
- Read carefully the whole subject before beginning.
- Watch out! This document could potentially change up to an hour before submission.
- This project will be corrected by humans only.
- This course is designed to build on previous days' concepts, try your hardest to finish everyday.
- Each day culminates in a portfolio piece, if you finish the day this is something you can use to get hired.
- When submitting, submit the folder of the Xcode project.
- Only the work submitted on the repository will be accounted for during peer-2-peer correction.
- Here is the [official manual of Swift](#) and the [Swift Standard Library](#)
- It is forbidden to use other libraries, packages, pods, etc. Unless otherwise stated in the project.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- You can discuss on the Piscine forum of your Intra!
- By Odin, by Thor! Use your brain!!!

Chapter III

Introduction

First we will be learning how to use Xcode, building things in swift without Xcode is possible since it's open source and works in linux, but for our purposes of building IOS, OSX and other Apple related apps it's not recommended as a viable solution at this time.

We will build some simple applications today learning how to use Xcode's Auto-layout, push notifications and API calls.

Each assignment was designed to build on the previous assignments knowledge and culminate in a portfolio piece. If we finish the day we will have a portfolio piece that reminds us to drink enough water everyday and will keep track of how much we've consumed.

Chapter IV

Exercise 00 : You Are Awesome (“Hello World”)

Exercise : 00
You Are Awesome
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit
Notes : n/a

For this first exercise we are learning to create a project, how to display text to the screen and how to make a button functional. The trick of this assignment is learning how to navigate Xcode.

Create a UIButton, in the main view, that when clicked will display a message in Xcode’s debug console and in a UILabel.

Chapter V

Exercise 01 : Buttons That Do Things

Exercise : 01
Buttons That Do Things
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit
Notes : n/a

Now it's not enough to have just one button, we need two. Also Apps should work in both landscape and portrait mode. Lucky for us swift/Xcode has tools like autolayout to help us.

Our app should have as follows:

1. Two buttons side by side that adapt to the rotation of the screen and change size relationally to the other elements of the page.
2. A UILabel with text describing how awesome you are inside of it, that changes between two fonts depending on which button was pressed.
3. A popup notification that alerts us to when a font was changed.

Pro Tip: Look up StackView, and see if you want to use it with autolayout.

Chapter VI

Exercise 02: Join The OO Party

Exercise : 02
Join The OO Party
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit (CAEmitterLayer / CAEmitterCell)
Notes : n/a

This assignment is to teach us the basics of Object Oriented Programming, or OOP/OO for short.

We are delving into some of the more interesting features of UIKit, dealing with layers, shaders and particles. Not only is this neat to know, but it's also a great chance to cut our teeth on how objects and inheritance works (which is foundational of what Object Oriented Programming is).

Use the project built previously, but now instead of pushing an update notification we are going to spray confetti all over the screen. You can make the confetti look however you want (snow, puppies, confetti, etc) but there must be 3 separate images, of varying sizes, and varying velocity. One button will create confetti, one button will stop it.

Congratulations, you now know how to use basic animations, use layers, particle systems and the core ideas of object oriented programming. This is worth celebrating.

Chapter VII

Exercise 03: Dates & Times

Exercise : 03
Dates & Times
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit
Notes : n/a

We are going to script an event to occur at a specific time, like a minimal alarm clock. A push notification will suffice.

Chapter VIII

Exercise 04: Stay Hydrated App

Exercise : 04
Stay Hydrated App
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit (CAEmitterLayer / CAEmitterCell)
Notes : n/a

Staying hydrated is important.

This assignment is to build an app that uses everything we've learned up until this point.

Our App Should:

- Use push notifications to remind you to drink water (on a timer).
- Keep track of how much water was drunk that day.
- Visually represent how much water that is in reference to the daily goal.
- Reset the amount at the end of the day

Chapter XI

Bonus : Adding Animations & Graphics

Bonus
Adding Animations & Graphics
Files to turn in: .xcodeproj and all necessary files
Allowed functions : Swift Standard Library, UIKit
Notes : n/a

Your stay hydrated app is your first IOS portfolio piece, make it look nice, give it the bells and whistles you wish it had, add some neat animations. Make it shine.