



Piscine iOS Swift - Day 05

Kanto

Maxime LEMORT mlemort@student.42.fr
42 Staff pedago@42.fr

*Summary: This document contains the subject for Day 05 from „Piscine iOS Swift“
from 42*

Contents

| | | |
|------|----------------------------------|----|
| I | Foreword | 2 |
| II | General Instructions | 3 |
| III | Introduction | 5 |
| IV | Exercice 00 : Tab Bar | 6 |
| V | Exercice 01 : Table View | 7 |
| VI | Exercice 02 : MapKit | 8 |
| VII | Exercice 03 : SegmentControlBar | 10 |
| VIII | Exercice 04 : Geolocalisation | 11 |
| IX | Exercice 05 : Location selection | 12 |
| X | Exercice 06 : Pin's color | 13 |

Chapter I

Foreword



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.
- The document can be relied upon, do not blindly trust the demos which can contain unrequired additions.
- You will have to submit one app every day (except for Day 01) on your git repository, submit the folder of the Xcode project.
- Here it is the official manual of [Swift](#) and of [Swift Standard Library](#)
- It is forbidden to use other libraries, packages, pods, etc. before Day 07
- 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 !!!



The videos on Intra were produced before Swift 3. Remove the prefix "NS" which you see in front of the class/struct/function in the code in the videos in order to use them in Swift 3.



Intra indicates the date and the hour of closing for your repositories. This date and hour also corresponds to the beginning of the peer-evaluation period for the corresponding piscine day. This peer-evaluation period lasts exactly 24h. After 24h passed, your missing peer grades will be completed with 0.

Chapter III

Introduction

Geolocation is an indispensable weapon of the good iOS developer kit, it is essential to know how to use it.

Apple makes available to you different frameworks like **MapKit** which allows you to use a very simple map or **CoreLocation** which allows you to manage the user's location.

These 2 frameworks will be your best friends for this day.

You will create a geolocation application of several places using:

- a **TabBarController** : to organise your different views.
- a **MKMapView** : for the map.
- a **CLLocationManager** : to geolocate the user.
- a **SegmentedControlBar** : to change the style of the map.
- a **MKAnnotationView** : to customize the pop-up map.

Chapter IV

Exercice 00 : Tab Bar

| | |
|---|---|
|  | Exercice : 00 |
| | Tab Bar |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit |
| | Notes : n/a |

Start by creating the project by choosing *Tabbed Application*.

Customize the icons of the 2 views already in the *MainStoryboard*.

One of the views will be the list of several places in a **table view**. The other will be the map on which the places will be displayed.

Chapter V

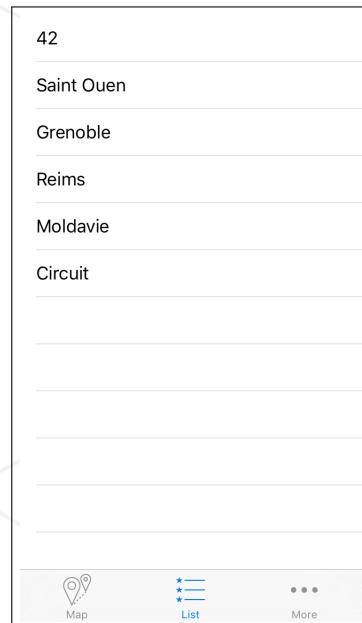
Exercice 01 : Table View

| | |
|---|---|
|  | Exercice : 01 |
| | Table View |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit |
| | Notes : n/a |

You will now add a **table view** to one of these 2 views. It must display at least 3 different places.



Think about organizing your data to make the development more easy in future exercises. You always have the right to add other files to your project.



42

Saint Ouen

Grenoble

Reims

Moldavie

Circuit

Map List More

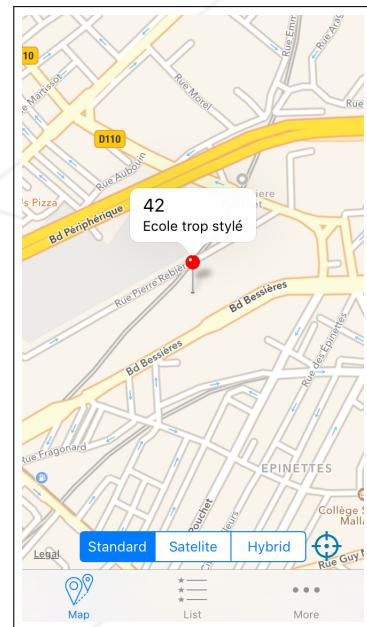
Chapter VI

Exercice 02 : MapKit

| | |
|---|---|
|  | Exercice : 02 |
| | MapKit |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit, MapKit |
| | Notes : n/a |

Let's get to the heart of the matter with this exercise, you will:

- Add a map to the second view.
- Display a *pin* to Ecole 42.
- Add a title and subtitle to this *pin*. This information must appear when you click on the pin.
- When it arrives on the map it must be zoomed on Ecole 42.



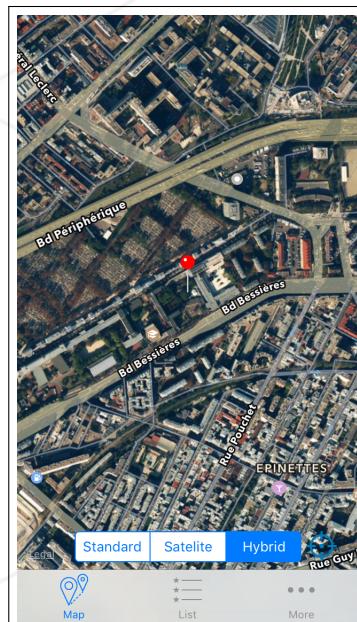
Chapter VII

Exercice 03 : SegmentControlBar

| | |
|--|---|
| | Exercice : 03 |
| | SegmentControlBar |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit, MapKit |
| | Notes : n/a |

Now that you have managed to display the map as you want, you can add a **segmented control bar** to select the mode of the map.

There are 3 ways to display a map : *Hybrid*, *Satellite* or *Standard*. The **segmented control bar** allow to change the mode at any moment.



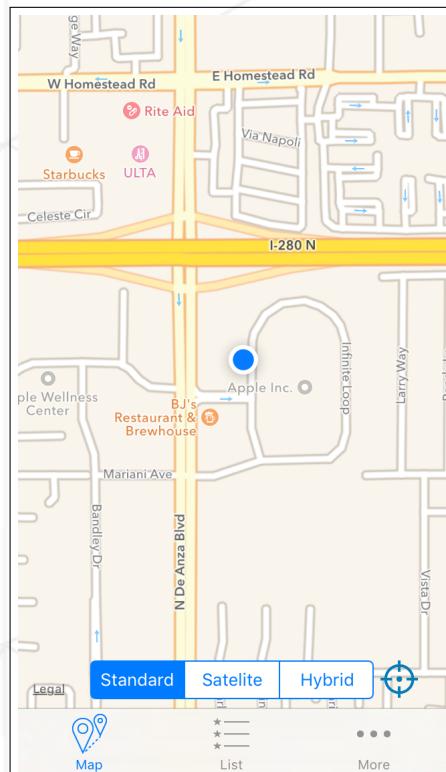
Chapter VIII

Exercice 04 : Geolocalisation

| | |
|--|---|
| | Exercice : 04 |
| | Geolocalisation |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit, MapKit, CoreLocation |
| | Notes : n/a |

Now that your map is functional, it would be nice if you could have a button to get geolocalized.

Add a button that should refocus the map to your position by adjusting the scale of the display to zoom in on you.



Chapter IX

Exercice 05 : Location selection

| | |
|---|---------------|
|  | Exercice : 05 |
| Location selection | |
| Files to turn in : .xcodeproj and all the necessary files | |
| Allowed functions : Swift Standard Library, UIKit, MapKit, CoreLocation | |
| Notes : n/a | |

You must now make your list functional by passing variables between your views :

- All the places in your list must be present on your view of the map as *pin* with a title and a subtitle.
- A click on one of the places on your list must return to the view of the map and zoom to the selected place.



Under no circumstances should you instantiate a controller and a view again.

Chapter X

Exercice 06 : Pin's color

| | |
|---|---|
|  | Exercice : 06 |
| | Pin's color |
| | Files to turn in : .xcodeproj and all the necessary files |
| | Allowed functions : Swift Standard Library, UIKit, MapKit, CoreLocation |
| | Notes : n/a |

You will now customize the *pin* of the map giving them different colors.

