



Piscine iOS Swift - Day 05

Kanto

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

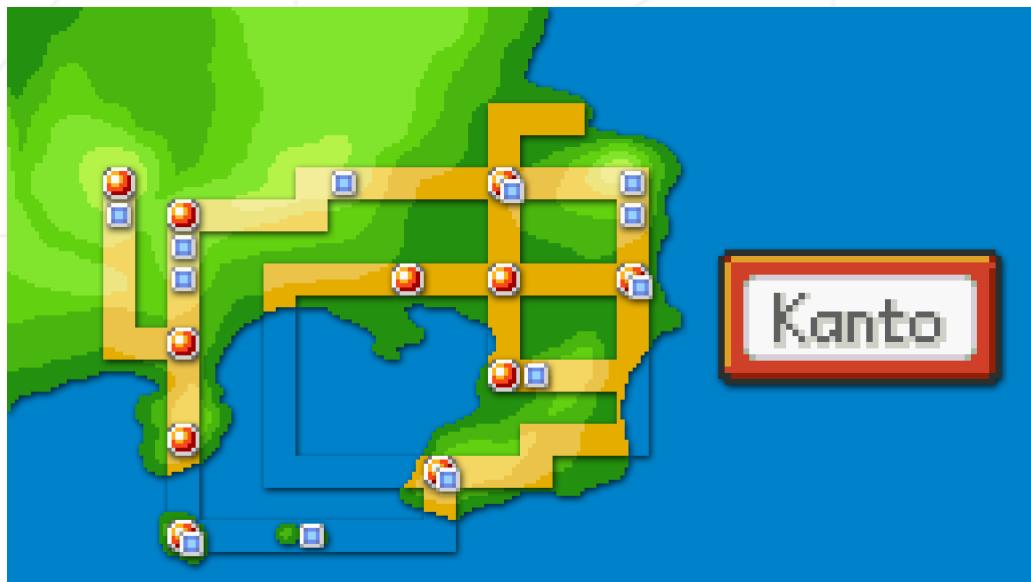
Résumé: Ce document contient le sujet du Day 05 de la piscine iOS Swift de [42](#)

Table des matières

I	Préambule	2
II	Consignes	3
III	Introduction	4
IV	Exercice 00 : Tab Bar	5
V	Exercice 01 : Table View	6
VI	Exercice 02 : MapKit	7
VII	Exercice 03 : SegmentControlBar	8
VIII	Exercice 04 : Geolocalisation	9
IX	Exercice 05 : Selection d'un lieu	10
X	Exercice 06 : Couleurs des Pins	11

Chapitre I

Préambule



Chapitre II

Consignes

- Seule cette page servira de référence : ne vous fiez pas aux bruits de couloir.
- Lisez attentivement l'intégralité du sujet avant de commencer.
- Le sujet peut changer jusqu'à une heure avant le rendu.
- Vos exercices seront corrigés par vos camarades de piscine.
- Le sujet fait foi, ne vous fiez pas toujours à la lettre aux demos qui peuvent contenir des ajouts supplémentaires non demandés.
- Vous devrez rendre une app par jour (sauf pour le Day 01) sur votre dépôt git, rendez le dossier du projet Xcode.
- Voici le manuel officiel de [Swift](#) et de [Swift Standard Library](#)
- Il est interdit d'utiliser d'autres bibliothèques, packages, pods... avant le Day 07
- Vous avez une question ? Demandez à votre voisin de droite. Sinon, essayez avec votre voisin de gauche.
- Pensez à discuter sur le forum Piscine de votre Intra !
- Réfléchissez. Par pitié, par Odin ! Nom d'une pipe.



Les vidéos sur l'intra ont été tournées avant Swift 3. Enlevez le prefix "NS" que vous voyez devant les class/struct/fonction dans le code des vidéos pour les utiliser en Swift 3.



L'intra indique la date et l'heure de fermeture de vos dépôts. Cette date et heure correspondent également au début de la période de peer-evaluation pour le jour de piscine correspondant. Cette période de peer-evaluation dure exactement 24h. Une fois ces 24h passées, vos notes peer manquantes seront complétées par des 0.

Chapitre III

Introduction

La géolocalisation est une arme indispensable du kit du bon développeur iOS, il est essentiel de savoir l'utiliser.

Apple met à votre disposition différents frameworks comme **MapKit** qui vous permet d'utiliser une map très simplement ou encore **CoreLocation** qui vous permet de gérer la localisation de l'utilisateur.

Ces 2 frameworks seront vos meilleurs amis pour cette journée.

Vous allez créer une application de géolocalisation de plusieurs lieux utilisant :

Un TabBarController : pour organiser vos différentes vues.

Une MKMapView : pour la map.

Un CLLocationManager : pour la géolocalisation de l'utilisateur.

Un SegmentedControlBar : pour changer le style de la map.

Des MKAnnotationView : pour customiser les pop up de la carte.

Chapitre IV

Exercice 00 : Tab Bar

	Exercice : 00
	Tab Bar
	Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires
	Fonctions Autorisées : Swift Standard Library, UIKit
	Remarques : n/a

Commencez par créer le projet en choisissant *Tabbed Application*.

Personnalisez les icônes des 2 vues déjà présentes dans le *MainStoryboard*.

Une des vues sera la liste de plusieurs lieux dans une **table view**. L'autre sera la map sur laquelle les lieux seront affichés.

Chapitre V

Exercice 01 : Table View

	Exercice : 01
	Table View
	Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires
	Fonctions Autorisées : Swift Standard Library, UIKit
	Remarques : n/a

Vous allez maintenant ajouter une **table view** à l'une de ces 2 vues. Elle devra afficher au moins 3 lieux différents.



Pensez à bien organiser vos données pour rendre le développement plus facile dans les prochains exercices. Vous avez toujours le droit d'ajouter d'autres fichiers à votre projet.

42

Saint Ouen

Grenoble

Reims

Moldavie

Circuit

 Map

 List

 More

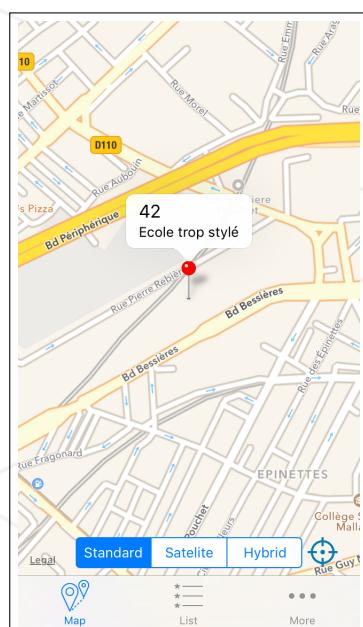
Chapitre VI

Exercice 02 : MapKit

	<p>Exercice : 02</p> <p>MapKit</p> <p>Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires</p> <p>Fonctions Autorisées : Swift Standard Library, UIKit, MapKit</p> <p>Remarques : n/a</p>
--	--

Rentrions dans le vif du sujet avec cet exercice, vous devrez :

- Ajouter une map dans la seconde vue.
- Afficher un *pin* sur l'école 42.
- Ajouter un titre et un sous titre à ce *pin*. Ces informations doivent apparaître lorsqu'on click sur le pin.
- Lorsqu'on arrive sur la map elle doit être zoomée sur l'école 42.



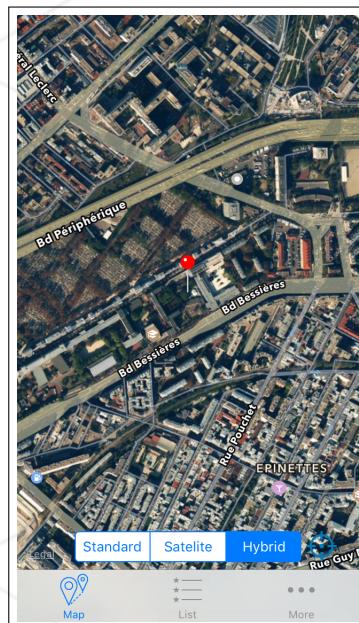
Chapitre VII

Exercice 03 : SegmentControlBar

	Exercice : 03
	SegmentControlBar
	Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires
	Fonctions Autorisées : Swift Standard Library, UIKit, MapKit
	Remarques : n/a

Maintenant que vous avez réussi à afficher la map comme vous le vouliez, vous pouvez rajouter un **segmented control bar** qui doit permettre de sélectionner le mode de la map.

Il existe 3 modes d'affichage de la map : *Hybrid*, *Satellite* ou *Standard*. Le **segmented control bar** doit permettre de changer de mode à tout moment.



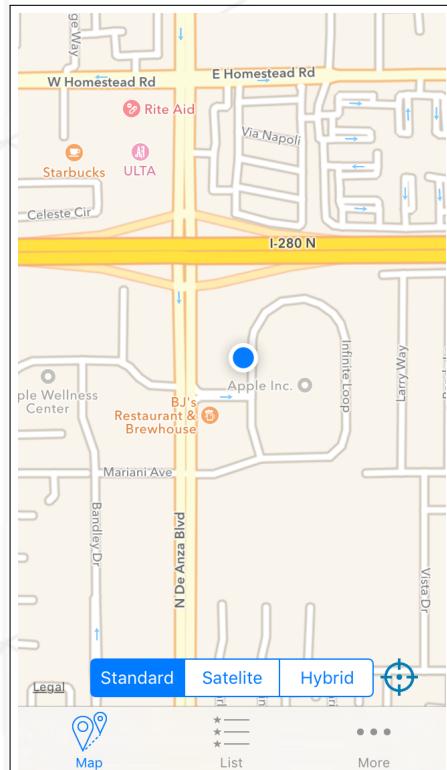
Chapitre VIII

Exercice 04 : Geolocalisation

	Exercice : 04
	Geolocalisation
	Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires
	Fonctions Autorisées : Swift Standard Library, UIKit, MapKit, CoreLocation
	Remarques : n/a

Maintenant que votre map est fonctionnelle, il serait bien que l'on puisse avoir un bouton pour se géolocaliser.

Ajoutez un bouton qui doit recentrer la map sur votre position en réglant l'échelle de l'affichage pour zoomer sur vous.



Chapitre IX

Exercice 05 : Selection d'un lieu

	Exercice : 05
	Selection d'un lieu
	Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires
	Fonctions Autorisées : Swift Standard Library, UIKit, MapKit, CoreLocation
	Remarques : n/a

Vous devez maintenant rendre votre liste fonctionnelle en passant des variables entre vos vues :

- L'ensemble des lieux de votre liste doivent être ainsi présents sur votre vue de la map sous forme de *pin* avec un titre et un sous-titre.
- Un clic sur un des lieux de votre liste doit renvoyer sur la vue de la map et zoomer sur le lieu sélectionné.



Vous ne devez en aucun cas instancier une nouvelle fois un controller et une vue.

Chapitre X

Exercice 06 : Couleurs des Pins

	Exercice : 06
Couleurs des Pins	
Fichiers à rendre : .xcodeproj et tous les fichiers nécessaires	
Fonctions Autorisées : Swift Standard Library, UIKit, MapKit, CoreLocation	
Remarques : n/a	

Vous allez maintenant personnaliser les *pin* de la map leur donnant différentes couleurs.

