



Around 42

iOS Introduction

Staff 42 piscine@42.fr

*Summary: This document serves to set an introduction to rush iOS.
@AlexisBarreyat and Mickaël Garcin .*

Contents

I	instructions	2
II	rendering process	3
III	Preamble	4
IV 00	Exercise: Creating the project	5
V	01 Activity: Launch Screen	6
VI	Exercise 02: Custom Icon	7
VII	Exercise 03: List	8
VIII	Exercise 04: Creation of map	10
IX	Exercise 05: Entracte	12
X	Exercise 06: A ffi Satellite chage	14
XI	Exercise 07: Segmented Control Bar	16
XII	Exercise 08: Geolocation	18
XIII	Exercise 09: Link list	20
XIV	Exercise 10: Bonus	22

Chapter I

Instructions

- Only this page serve as a reference: Do not f i rm the hallway noise.
- You must follow the rendering process for all your exercises.
- Your exercises will be corrected by your pool mates.
- You have a question ? Ask your neighbor to the right. Otherwise, try your left neighbor.
- Your reference manual called Google.

chapter II

rendering process

- Your rendering: votrelogin.zip containing all the folders on your xcode project.
- The purpose of this rush is to achieve a iphone application that contains:
 - A loading screen
 - A list of defined places
 - A map on which are present the elements of the list
- The images and colors are not imposed.

Chapter III

Preamble

Enigma /e.nigm/ female Etymology: From the Latin *aenigma*.

mind game that test reflection party who must answer to a query whose meaning is hidden instead to get on with more important tasks such as rush example. But who can however be extremely useful for the future.



chapter IV

00 Exercise: Creating the project

	Exercise: Creating the
	Project 00
Render Folder: <i>ex 00 / Files to make:</i>	
Functions Authorized: Any	
Remarks: n / A	

- Start by creating the project in xcode.
- Choose Tabbed Application.
- Select good options "Swift" and "iPhone".
- Test for functionality by launching the project in the simulator.
- Your project must operate at least on iPhone 6.



Tabbed Application lets start with an existing design with multiple views.

chapter V

01 Activity: Launch Screen

	Activity: 01 Launch
	Screen
Render Folder: ex 01 / files to make:	
Functions Permitted: Any	
Remarks: n / A	

- Now we attack at the sight of load application.
- You must customize your loading screen so he ffi che:
 - Custom text that is not just the name of your project.
 - A picture.
- Check that everything works by running the simulator.



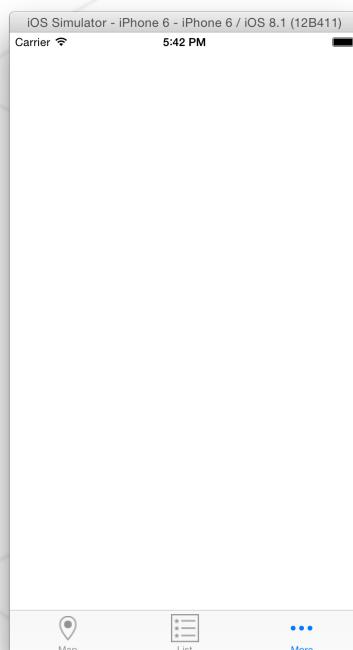
Look at the auto-size, auto-layout and constraints.

chapter VI

Exercise 02: Custom Icon

	Exercise: 02
	Custom Icon
Render Folder: ex 02 / Files to make:	
Functions Authorized: Any	
Remarks: n / A	

- Create a new view in the storyboard.
- You must customize this view to the icon of this view is customized (for example using one that is given to you with the subject).



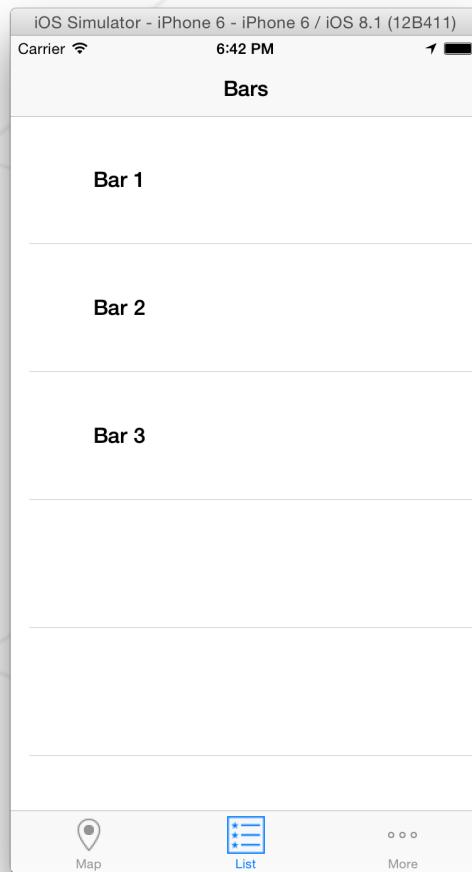
chapter VII

Exercise 03: List

	Exercise: 03
	Listing
Render Folder: ex 03 / Files to make:	
Functions Authorized: Any	
Remarks: n / A	

We go here a ffl expensive a list of places that you will use on a map. So choose specific locations in order to customize your application.

- Use an existing view the Storyboard for a ffl an expensive Tableview.
- Add several items ahead of your own.



To do this, go you have to follow these steps:

- Declaring UITableView controller in your view.
- Add the necessary protocols to class in his statement.
- Declare several methods for dividing the table into sections, giving the index of the cell, and select them.

There you go ! It was simple right?



If you do not have any monitoring, you can always find help
[right here](#) and [right here](#) or [right here](#) but certainly not [right here](#).

chapter VIII

Exercise 04: Create the map

	Exercise: 04
	Listing
Render Folder:	<i>ex 04 / Files to make:</i>
Functions Authorized:	Any
Remarks:	n / A

- In a second view, import the MapKit the controller of sight.
- Add a map.



chapter IX

Exercise 05: Intermission

A autostereogram consists of a single image that gives the illusion of a three-dimensional scene from a two-dimensional image.

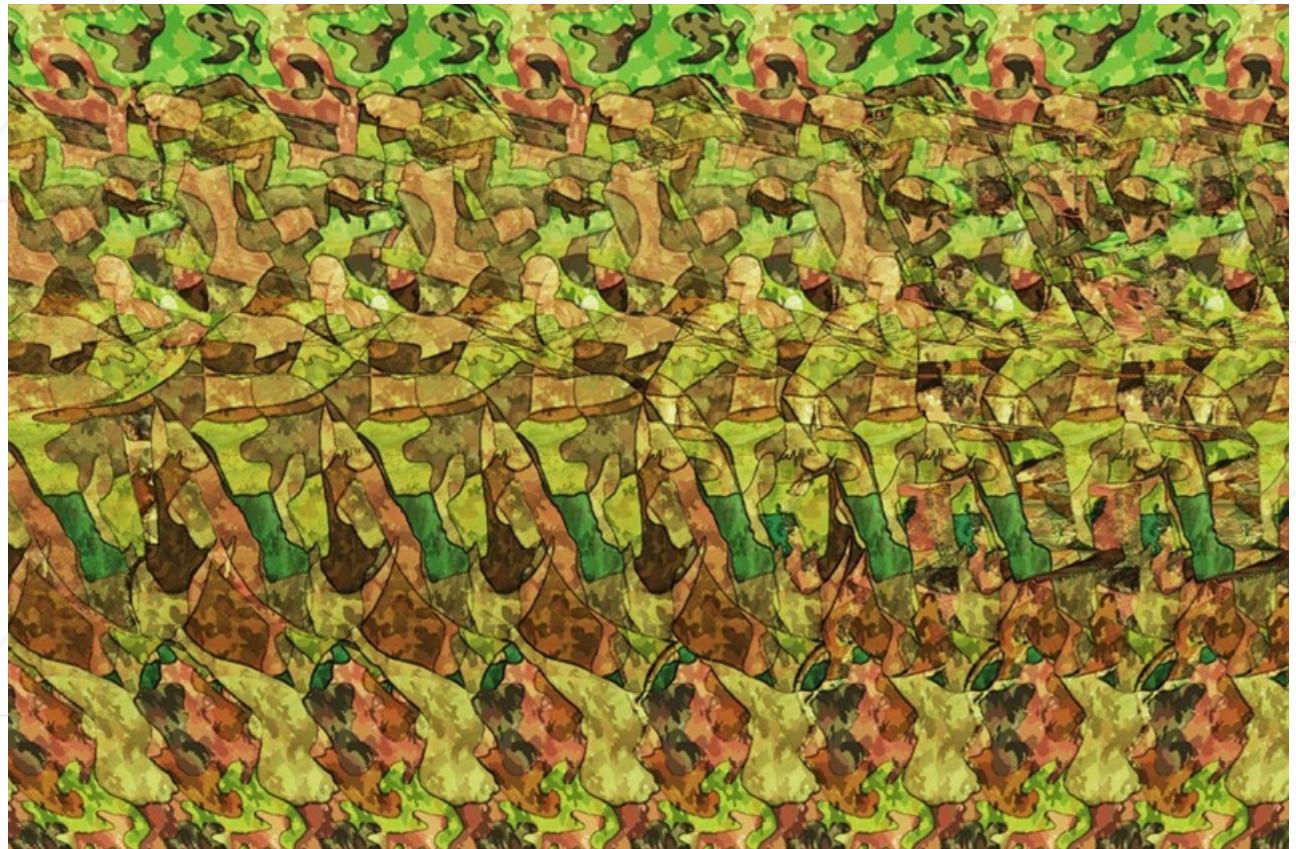
In order to perceive 3D shapes through autostereograms, the brain must make an ocular convergence effort and focus dissociated of accommodation.

Unlike conventional stereo views, the autostereograms can be viewed without a stereoscope.

A stereoscope has two 2D images of an object taken from a slightly different angle, so that the brain has a 3D view of this object.

In the case of autostereogram, the brain receives two 2D patterns in each eye, but can not match them correctly.

It assembles the units in a virtual object because of bad angles parallax, and the virtual object is therefore not on the same plane as the image of the autostereogram.



- Move closer to your screen by securing a point that would be located behind the screen.
- Move away slowly, without trying to change focus.
- You should see a 3D scene with among other animals.
- Remember that animal.

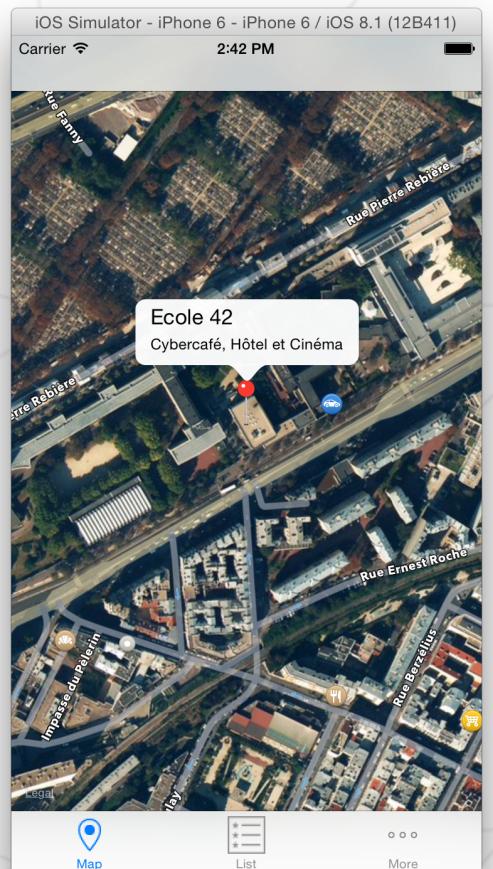
chapter X

Exercise 06: A ffi Satellite chage

	Exercise: 06
	Listing
Render Folder:	ex 06 / Files to make:
Functions Authorized:	Any
Remarks:	n / A

- Now a ffi at the map in "Hybrid" mode, or "Satellite" (rather than the default "Standard").
- A ffi in a pine of 42.
- Modifi rm the default scale of a ffi drying in order to make it appear zoomed in school.

- Add a title and a subtitle.



chapter XI

Exercise 07: Segmented Control Bar

	Exercise: 07 Segmented
	Control Bar
Render Folder:	ex 07 / Files to make:
Functions Authorized:	Any
Remarks:	n / A

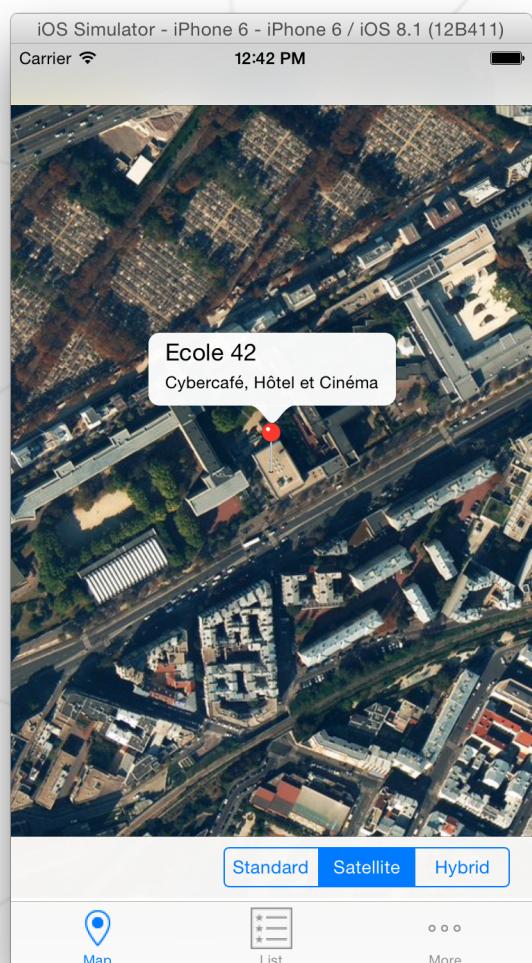
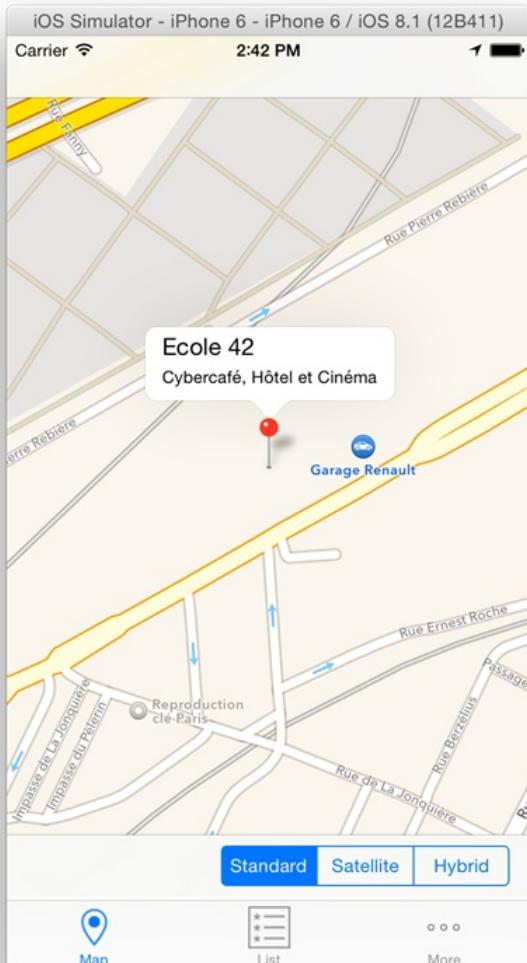
Now that you have successfully priced a ffi the map as you like, you can add buttons that should allow to select your preferred view. So you need to do:

- a button for each mode of the map.
- rendering function of these buttons.



Search the Segmented Control Bar of the side as the fiscal year indicated.

Your made should look something like this:



chapter XII

Exercise 08: Geolocation

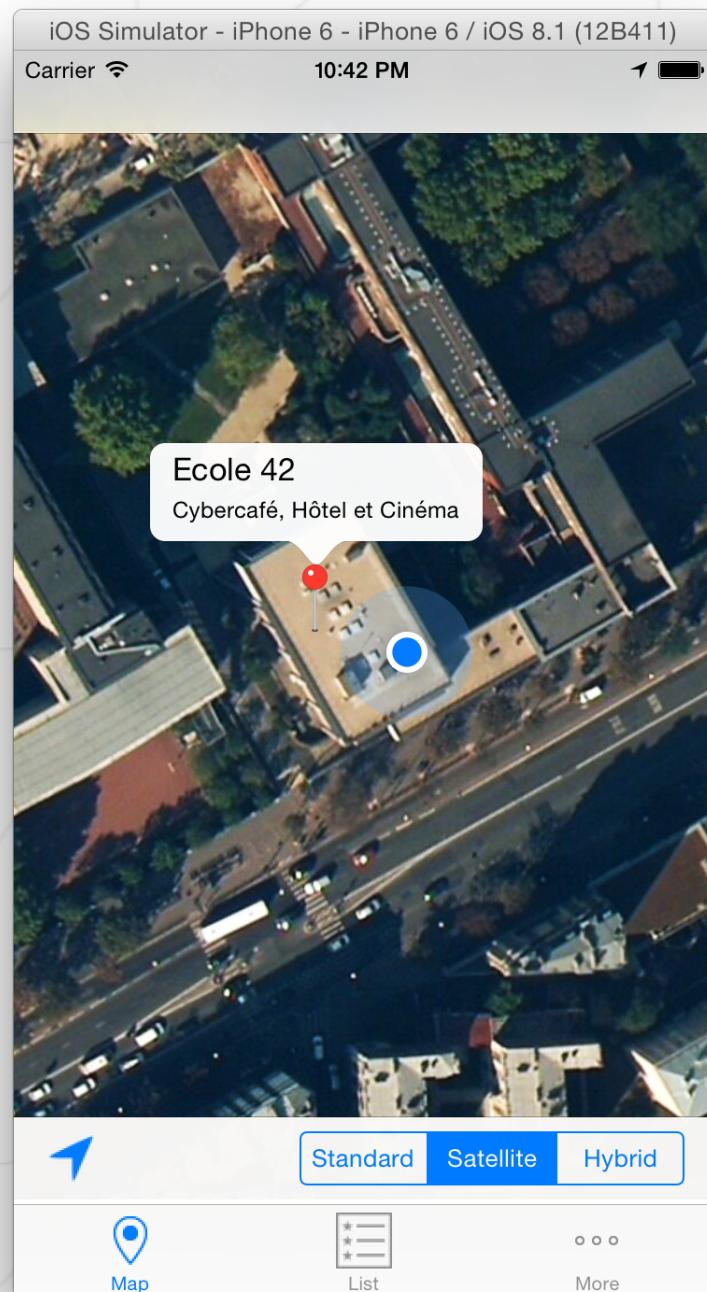
	Exercise: 08
	Geolocation
Render Folder: ex 08 / Files to make:	
Functions Authorized: Any	
Remarks: n / A	

Now that your map is functional. It would be nice if we can have a button to geotag. So you need to do:

- A superb active geolocation button that should refocus the map on your location.
- A **required balboa** would be to readjust the scale of a ffi chage to zoom in on your position.



Of course, your location will be virtual and not real for this test.



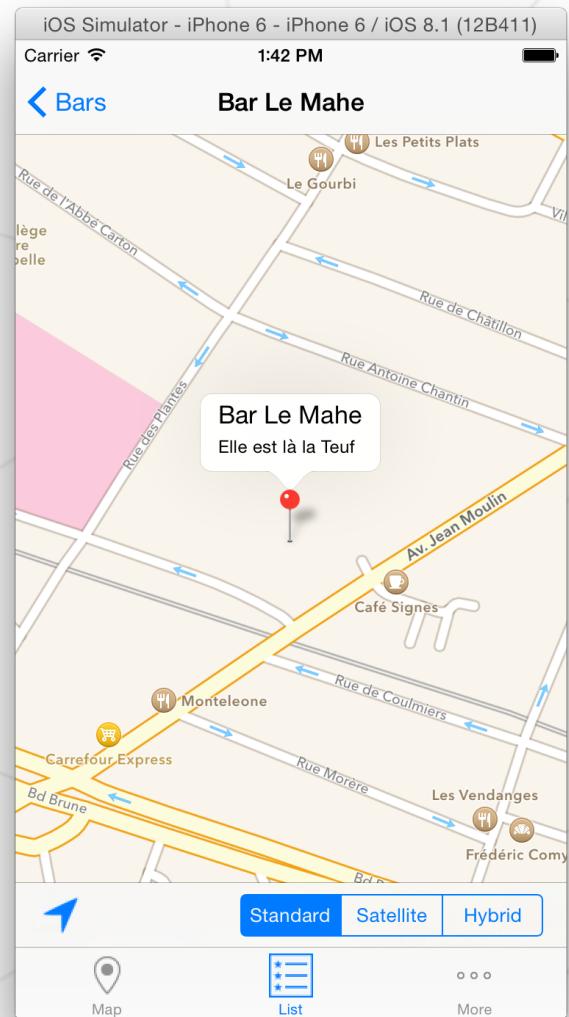
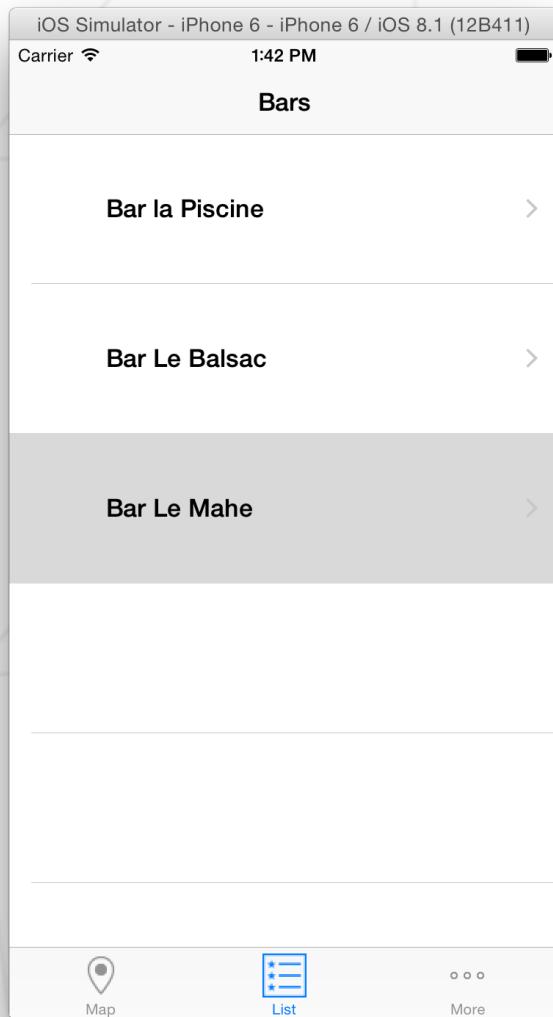
chapter XIII

Exercise 09: Link list

	Exercise: 09
	Link list
Render Folder: ex 09 / Files to make:	
Functions Authorized: Any	
Remarks: n / A	

Now you must make your list Functional passing variables between your views:

- All places of your list should be as present on your map view. (Thus recover the data from your list)
- Clicking on one of the places on your list should return on a new detailed view showing a map with only the location of the element in question and the information related to it.



The duplication of data is prohibited. They should only be declared once. All views are therefore affected when they are modified.

chapter XIV

Exercise 10: Bonus

	Exercise: 10
	bonus
Render Folder: ex 10 / Files to make:	
Functions Authorized: Any	
Remarks: n / A	

Bonuses are free. You can therefore include, among others:

- A settings page (with different effective settings)
- An intuitive and attractive interface.
- different pine colors in different places (or types of places).
- advanced pictures with descriptions.
- An inclusion of the animal found in the intermission somewhere in your application.
- Custom icons for your buttons.
- Having added somewhere in your app the answer to the riddle.