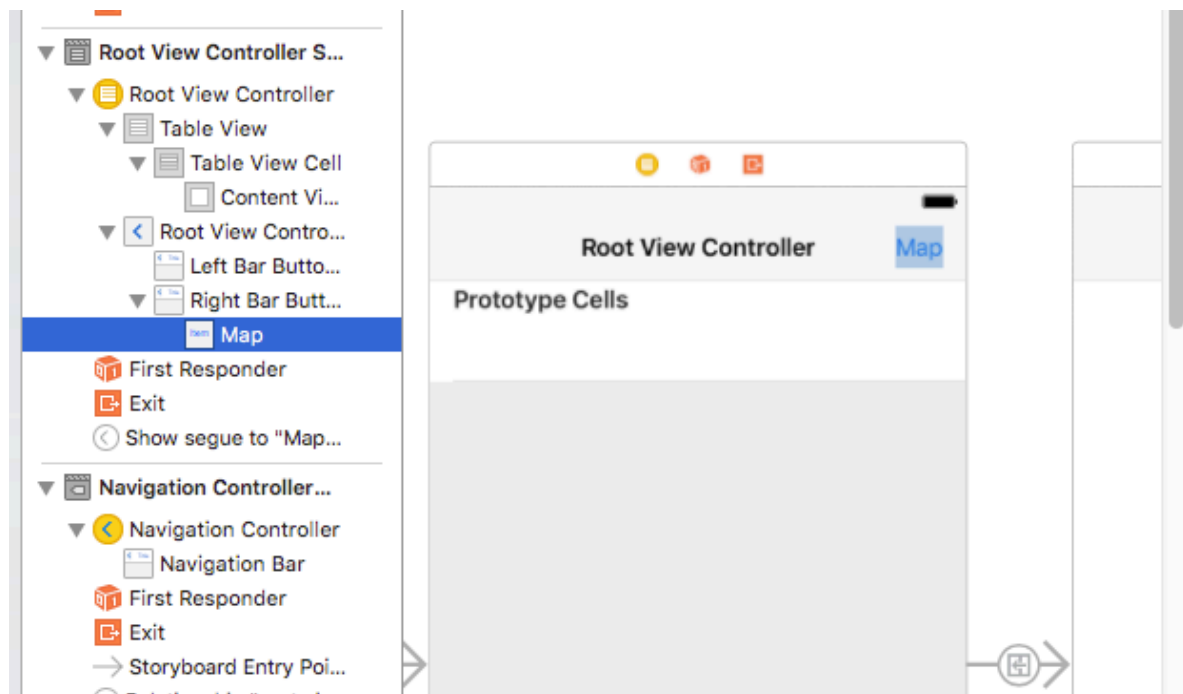


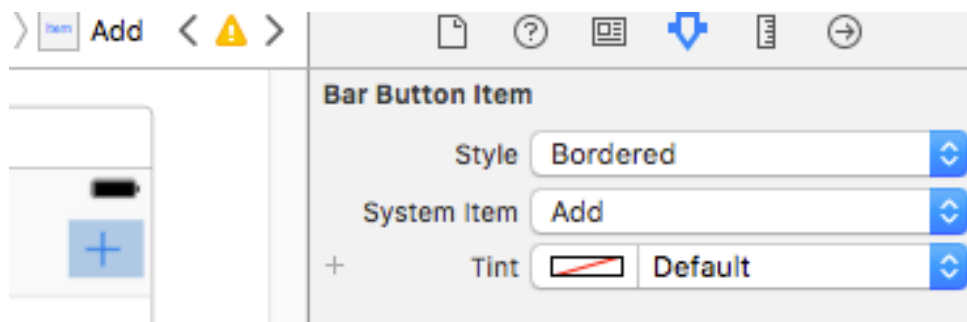
Memorable Places App

We want to write an app that contains an editable list of geographical places. By clicking a specific item in the list a map is displayed centered on that place. By long tapping a place onto the map, the corresponding place is added to the list.

1. Create a new project, name it Memorable Places
2. Add a Navigation Controller to the storyboard
3. Set the Navigation Controller as the initial controller
4. Set the title "Map View Controller" on the View Controller
5. Add a Bar Button Item on the Navigation Table View (Right side), call it "Map" and make it triggering a show segue to the Map View



6. Build and Run
7. Change the Map Bar Button symbol to a "+"



8. Add a Map Kit View object in the Map View Controller. Set all-screen constraints
9. Open ViewController.swift, the class file of the MapViewController, import

MapKit and add a outlet for the map view

10. Create a Cocoa Touch File class TableViewController.swift (subclass of UITableViewController), and associate the Root View Controller to it from the Identity pane. Then return 1 section and 3 cells for section in the relevant class methods definition. Uncomment the cellForRowAtIndexPath method, set the cell's titleLabel text to "ciao" and the cellIdentifier to "Cell". Also in the Attribute Inspector of the Table View Cell set the Identifier field to "Cell"
11. Open the class file of the MapViewController and add a outlet for the map view
12. Add support to the CoreLocation framework (build settings and plist keys for authorisation)
13. In ViewController viewDidLoad create a CLLocationManager object, initialise it and start the location update task
14. then create a 2 sec long lasting UI long pressure gesture recogniser

```
29 //long pressure gesture recognizer allows to add a new memorable place to the list
30 let uilgr = UILongPressGestureRecognizer(target: self, action:
    #selector(ViewController.action(_:)))
31 uilgr.minimumPressDuration = 2.0
32 map.addGestureRecognizer(uilgr)
```

15. On gesture action add a map annotation

```
54 func action(_ gestureRecognizer: UIGestureRecognizer) {
55     if gestureRecognizer.state == UIGestureRecognizerState.began {
56         let touchPoint = gestureRecognizer.location(in: self.map)
57         let coordinateOnMap: CLLocationCoordinate2D = map.convert(touchPoint, toCoordinateFrom: self.map)
58         let annotation = MKPointAnnotation()
59         annotation.coordinate = coordinateOnMap
```

16. the annotation title should contain the address of the user location.

17. We will extract this information by reverse geocoding

```
61 let locationOnMap = CLLocation(latitude: coordinateOnMap.latitude, longitude: coordinateOnMap.longitude)
62
63 //convert location in placemark by inverse geocoding
64 CLGeocoder().reverseGeocodeLocation(locationOnMap, completionHandler: {(placemarks, error) -> Void in
65     var title: String = "On \(NSDate())"
66     if (error != nil) {
67         print(error)
68     } else {
69         //get the placemark data
70         if let p = placemarks?[0] {
71             var subThoroughfare: String = ""
72             var thoroughfare: String = ""
73             if p.subThoroughfare != nil {
74                 subThoroughfare = p.subThoroughfare!
75             }
76             if p.thoroughfare != nil {
77                 thoroughfare = p.thoroughfare!
78                 title = "\(subThoroughfare) \(thoroughfare)"
79             }
80         }
81     }
82 }
```

18. and the write down it as annotation title

```
81 //write down the placemark data as annotation title
82 annotation.title = title
83 annotation.subtitle = "added as new memorable place"
84 self.map.addAnnotation(annotation)
85 }
86 }
```

19. In TableViewController.swift declare a global dictionary with stain:string key-value items, to store latitude,longitude and description of each of the new memorable places we are going to add

```

9   import UIKit
10
11   var places = [Dictionary<String,String>()]

```

20. update the number of cells to the dictionary count

```

35   override func tableView(_ tableView: UITableView, numberOfRowsInSectionSection section: Int) -> Int {
36       return places.count
37   }

```

21. and add the reloadData call to make this happen when we came back to this scene

```

51   //this method forces the TVC to refresh the cell content right before it is showed up
52   override func viewWillAppear(_ animated: Bool) {
53       tableView.reloadData()
54   }

```

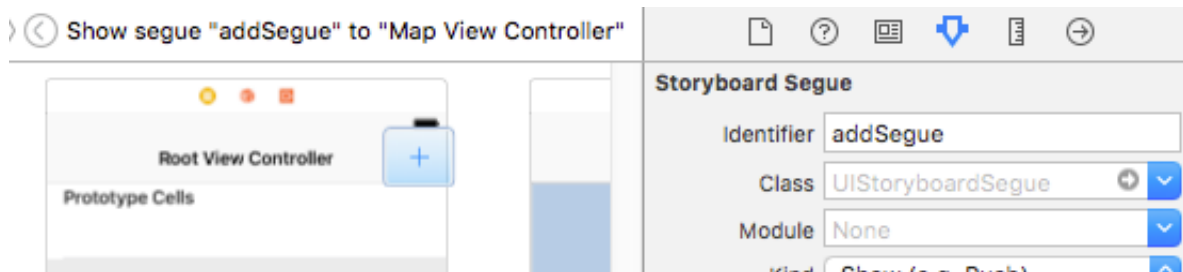
22. In ViewController.swift add a new item to the dictionary as soon as the placemark data have been managed

```

80   }
81   //add a new item to the list of memorable places
82   places.append(["Name":title,"Lat": "\(coordinateOnMap.latitude)", "Lon": "\(coordinateOnMap.longitude)"])

```

23. Now we want to center the map on the specific location we tapped in the TVC list. However we could reach the Map screen also through the navigation "+" Bur Button. In that case we would like to center the map on the current user's location. To do that we have to use the prepareForSegue in the source scene (TableViewCell.swift), where we set to -1 the global variable activePlace if the segue identifier matches the "addSegue" specified in the storyboard



```

57   override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
58       if segue.identifier! == "addSegue" {
59           activePlace = -1
60       }
61   }

```

24. Next challenges: add permanent storage of the memorable place list and the delete function on the list items