Step 1 - Continued from Part II

1. Open the app that you completed in Part II. Alternatively, open the app as it is found in the folder “DataBasedApp-Part3Beginning”. This is your starting point.

Step 2 - Add the MapKit Framework

2. To use any mapping abilities, you must include the MapKit Framework to the project. There is a Frameworks Group in the Project. To add a Framework, select the project in the navigator, and then select “Build Phases”.
3. Expand the section “Link Binary With Libraries”.
4. Click on the “+” button and type “MapKit”.
5. Expand the “iOS 5.0” folder, if necessary, and select *MapKit.framework.*
6. This will add the MapKit framework to the root of the project. You can drag it to the Frameworks group for better organization.

Step 3 - Create a ViewController for the Map View

7. Create a new UIViewController, with XIB file, called *MapViewController.*
8. Add a MapView view as a child of the main view for the newly create XIB file.
9. Create a new referencing outlet for the MapView object and call it *mapView*. This should be done in the File’s Owner class .h file (MapViewController.h).

Step 4 - Display the MapView Controller

10. Let’s add a button to the navigationItem of the CityInformationViewController.
11. Open *CityInformationViewController.m*, and add the following code to the viewDidLoad selector. This will create a new button item with code, and set the property of the navigationItem to that new button.

```
UIBarButtonItem *mapButton = [[UIBarButtonItem alloc] initWithTitle:@"Map"
style:UIBarButtonItemStylePlain target:self action:@selector(mapButtonAction:)];

self.navigationItem.rightBarButtonItem = mapButton;
```

12. Notice that we reference a selector called *mapButtonAction*. We need create a selector in that same class file called *mapButtonAction*.

```
-(void)mapButtonAction:(id)sender{
}
```

13. In that selector, add some code to create a new MapViewController and push it to the

navigationController. You can find the code to do this in the “*didSelectRowAtIndexPath*” selector in the EMTableViewController.m class.

NOTE: Don’t forget to import the *“MapViewController.h”* file into the class you’re working with.

RUN: If you run the app now, you will see an ERROR!

14. You need to import the MapKit Framework into *MapViewController.h* file.

```
#import <MapKit/MapKit.h>
```

RUN: Running the app at this point should show the new button on the navigation bar, and when you click on it you should see a map appear.

Step 3 - Give the map some coordinates

15. Add a new property to the “*City”* class. Call it “*location”* and make it a CLLocationCoordinate2D type.

```
@property (nonatomic) CLLocationCoordinate2D *location;
```

NOTE: You will need to import the MapKit Framework to the City.h file.
NOTE: Don’t forget to synthesize your newly created property.

16.