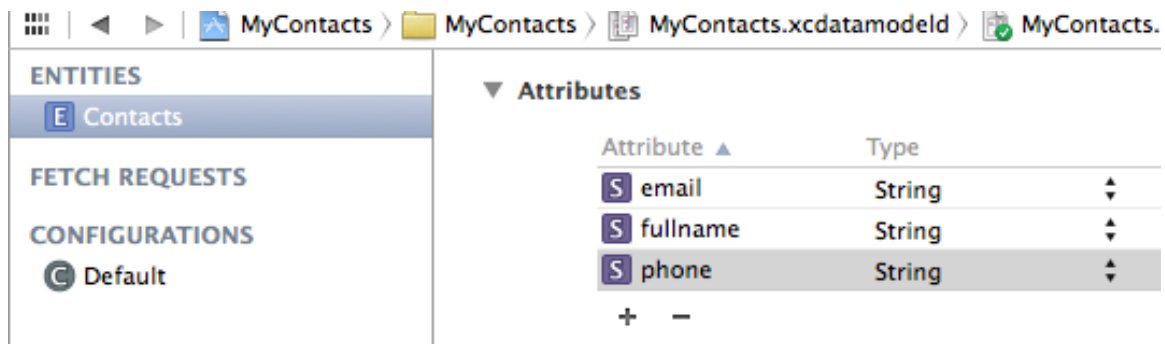


MyContacts database using CoreData. Insert, Update, and Delete

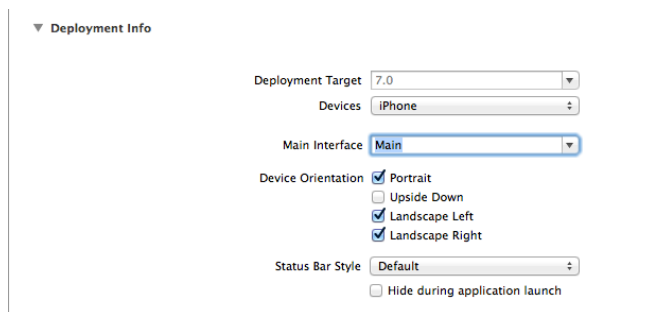
- 1) Create New **Empty Application** name **MyContacts** and check **Use Core Data**
- 2) Select CoreData **MyContacts.xcdatamodeld** from the Project Navigator
- 3) Add Entity to Core Data
 - a. Click the **+ Add Entity** button
 - b. Double click **Entity** under **Entities** and rename to **Contacts**
- 4) Add Attributes to Entity **Contacts**
 - a. Click the **+** symbol under **Attributes** and add the following attribute, type

Attribute, Type

- b. fullname, string
- c. email, string
- d. phone, string



- 5) Add StoryBoard
 - a. Right-click on MyContacts project name
 - b. Select **New File > User Interface**
 - c. Select **Storyboard**
 - d. Save as **Main**
 - e. Select **Create**
- 6) Select Main Interface
 - a. Select **MyContacts** under project navigator
 - b. Select under **Deployment Info** the dropdown **Main Interface** and select **Main.storyboard**

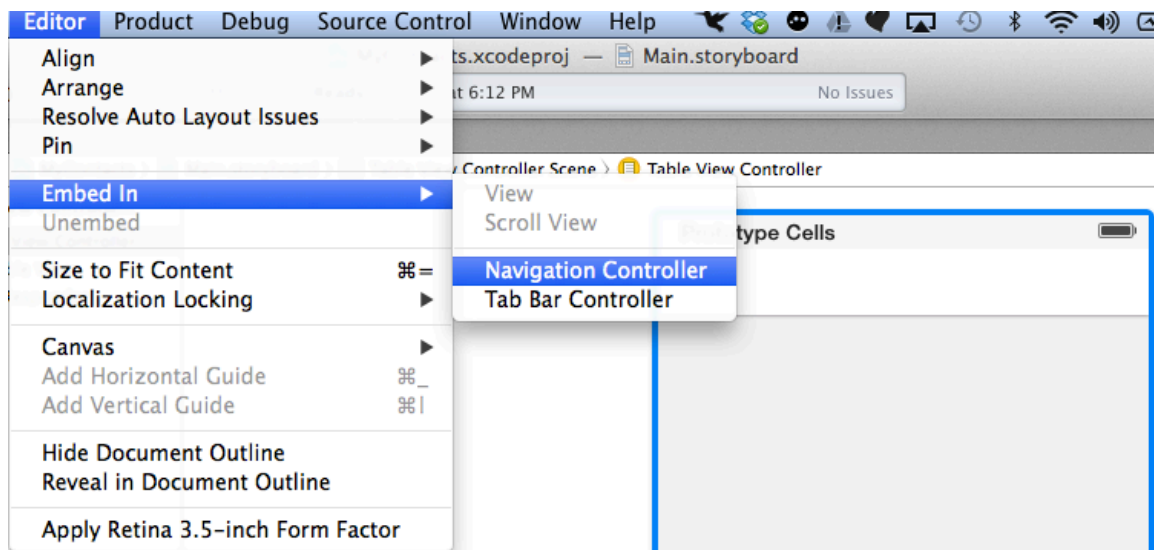


- 7) Select **MyContactsAppDelegate.m** and edit **launchOptions**

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
    // Override point for customization after application launch.
    // self.window.backgroundColor = [UIColor whiteColor];
    // [self.window makeKeyAndVisible];
    return YES;
}
```

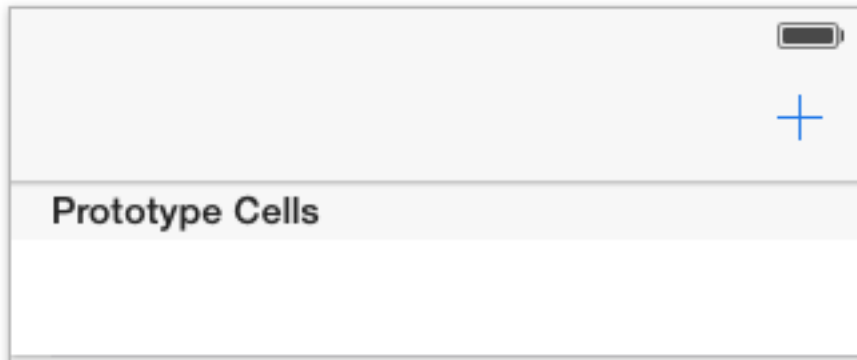
- 8) Add TableViewController
 - a. Select **Main.storyboard**

- b. Add **TableViewController**
- c. Select **TableViewController**
- d. Embed in **Navigation Controller**
 - i. **Editor > Embed In > Navigation Controller**
 - ii. **Save**
- e. Save



- 9) Add **New File > Objective-C Class > Next**
 - a. Name Class: **MyContactsViewController**
 - b. Select Subclass: **UITableViewController**
 - c. Select **Next**
 - d. Select **Create**
- 10) Add class to **MyContactsViewController**
 - a. Select **Main.storyboard**
 - b. Select **TableViewController**
 - c. Select **Identity Inspector**
 - i. Select Dropdown
 - ii. Select class **MyContactsViewController**
 - iii. **Save**
- 11) Add new **ViewController**
 - a. Select **Main.storyboard**
 - b. Add **ViewController**
- 12) Add **New File > Objective-C Class > Next**
 - a. Name Class: **MyDetailViewController**
 - b. Select Subclass: **UIViewController**
 - c. Select **Next**
 - d. Select **Create**
- 13) Add class to **MyDetailViewController**
 - a. Select **Main.storyboard**
 - b. Select **ViewController**
 - c. Select **Identity Inspector**
 - i. Select Dropdown
 - ii. Select class **MyDetailViewController**
 - iii. **Save**

- 14) Add Button to right side of Navigation item bar
- Select Bar Button item from outline view
 - Select Attribute **Type**
 - Style: **Bordered**
 - Identifier: **Add**

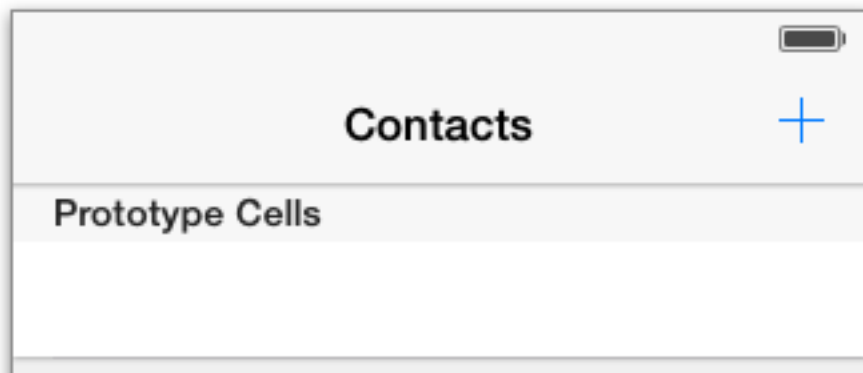


15) Add Navigation Bar to **MyDetailViewController** under battery icon

16) Add 2 buttons Navigation Bar

- Left button title **Back**
 - Style: **Bordered**
 - Identifier: **Add**
- Right button title **Save**
 - Style: **Bordered**
 - Identifier: **Add**

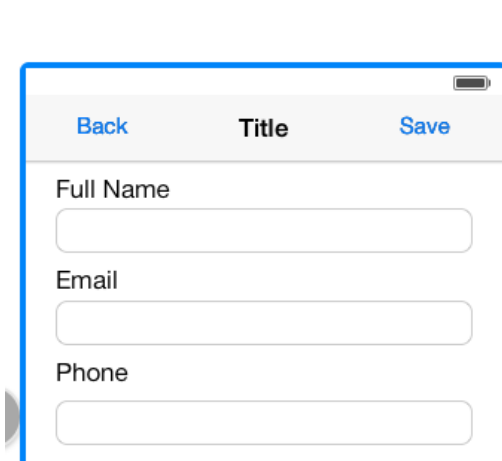
17) Select MyContactsViewController from StoryBoard and set Title of Navigation item bar **Contacts**



18) Add controls to **MyDetailViewController** (make look like below)

- 3 Labels**
- 3 TextFields**

c. Add **outlets** and **actions**



```
//  
// MyDetailViewController.h  
// MyContacts  
//  
// Created by Charles Konkol on 11/10/13.  
// Copyright (c) 2013 Chuck Konkol. All rights reserved.  
//  
  
#import <UIKit/UIKit.h>  
  
@interface MyDetailViewController : UIViewController  
⦿ @property (strong, nonatomic) IBOutlet UITextField *fullname;  
⦿ @property (strong, nonatomic) IBOutlet UITextField *email;  
⦿ @property (strong, nonatomic) IBOutlet UITextField *phone;  
⦿ - (IBAction)btnSave:(id)sender;  
⦿ - (IBAction)btnBack:(id)sender;  
  
@end
```

19) Control drag + button on **MyDetailViewController** to **MyContactsViewController** and select Modal

20) Select **Table View Cell** from **MyContactsViewController**

- a. Style: **Right Detail**
- b. Identifier: **Cell**
- c. Click on the **Modal segue from Cell to My Detail**
- d. Change identifier to **UpdateContacts**

21) **Modify MyDetailViewController.h** Add before @end

```
@property (strong) NSManagedObjectContext *contactdb;
```

22)Modify MyDetailViewController.m

Replace code from `@implementation` MyDetailViewController to `@end`

```
@synthesize contactdb;
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.
    if (self.contactdb) {
        [self.fullname setText:[self.contactdb valueForKey:@"fullname"]];
        [self.email setText:[self.contactdb valueForKey:@"email"]];
        [self.phone setText:[self.contactdb valueForKey:@"phone"]];
    }
}
- (NSManagedObjectContext *)managedObjectContext {
    NSManagedObjectContext *context = nil;
    id delegate = [[UIApplication sharedApplication] delegate];
    if ([delegate respondsToSelector:@selector(managedObjectContext)] {
        context = [delegate managedObjectContext];
    }
    return context;
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}
- (IBAction)btnsave:(id)sender {
    NSManagedObjectContext *context = [self managedObjectContext];

    if (self.contactdb) {
        // Update existing device
        [self.contactdb setValue:self.fullname.text forKey:@"fullname"];
        [self.contactdb setValue:self.email.text forKey:@"email"];
        [self.contactdb setValue:self.phone.text forKey:@"phone"];
    } else {
        // Create a new device
        NSManagedObject *newDevice = [NSEntityDescription
insertNewObjectForEntityForName:@"Contacts" inManagedObjectContext:context];
        [newDevice setValue:self.fullname.text forKey:@"fullname"];
        [newDevice setValue:self.email.text forKey:@"email"];
        [newDevice setValue:self.phone.text forKey:@"phone"];
    }
    NSError *error = nil;
    // Save the object to persistent store
    if (![context save:&error]) {
        NSLog(@"Can't Save! %@ %@", error, [error localizedDescription]);
    }

    [self dismissViewControllerAnimated:YES completion:nil];
}
- (IBAction)btnCancel:(id)sender {
    [self dismissViewControllerAnimated:YES completion:nil];
}
```

```

23)Modify MyDetailViewController.h Add after @interface
    @property (strong) NSMutableArray *contactarray;
24)Modify MyDetailViewController.m
    a. #import "MyDetailViewController.h"
    b. Replace code below @implementation MyContactsViewController to @end
- (NSManagedObjectContext *)managedObjectContext
{
    NSManagedObjectContext *context = nil;
    id delegate = [[UIApplication sharedApplication] delegate];
    if ([delegate respondsToSelector:@selector(managedObjectContext)]) {
        context = [delegate managedObjectContext];
    }
    return context;
}
- (id)initWithStyle:(UITableViewStyle)style
{
    self = [super initWithStyle:style];
    if (self) {
        // Custom initialization
    }
    return self;
}
- (void)viewDidLoad
{
    [super viewDidLoad];

}
- (void)viewDidAppear:(BOOL)animated
{
    [super viewDidAppear:animated];

    // Fetch the devices from persistent data store
    NSManagedObjectContext *managedObjectContext = [self managedObjectContext];
    NSFetchRequest *fetchRequest = [[NSFetchRequest alloc]
initWithEntityName:@"Contacts"];
    self.contactarray = [[managedObjectContext executeFetchRequest:fetchRequest
error:nil] mutableCopy];

    [self.tableView reloadData];
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}
#pragma mark - Table view data source
- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
{
    return 1;
}

```

```

- (NSInteger)tableView:(UITableView *)tableView
numberOfRowsInSection:(NSInteger)section
{
    return self.contactarray.count;
}

- (UITableViewCell *)tableView:(UITableView *)tableView
cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString *CellIdentifier = @"Cell";
    UITableViewCell *cell = [tableView
dequeueReusableCellWithIdentifier:CellIdentifier forIndexPath:indexPath];

    // Configure the cell...
    NSManagedObject *device = [self.contactarray objectAtIndex:indexPath.row];
    [cell.textLabel setText:[NSString stringWithFormat:@"%s", [device
valueForKey:@"fullname"], [device valueForKey:@"email"]]];
    [cell.detailTextLabel setText:[device valueForKey:@"phone"]];
    return cell;
}

// Override to support conditional editing of the table view.
- (BOOL)tableView:(UITableView *)tableView canEditRowAtIndexPath:(NSIndexPath
*)indexPath
{
    // Return NO if you do not want the specified item to be editable.
    return YES;
}

// Override to support editing the table view.
- (void)tableView:(UITableView *)tableView
commitEditingStyle:(UITableViewCellEditingStyle)editingStyle
forRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSManagedObjectContext *context = [self managedObjectContext];
    if (editingStyle == UITableViewCellEditingStyleDelete) {
        // Delete object from database
        [context deleteObject:[self.contactarray objectAtIndex:indexPath.row]];

        NSError *error = nil;
        if (![context save:&error]) {
            NSLog(@"Can't Delete! %s", error, [error localizedDescription]);
            return;
        }

        // Remove device from table view
        [self.contactarray removeObjectAtIndex:indexPath.row];
        [self.tableView deleteRowsAtIndexPaths:[NSArray arrayWithObject:indexPath]
withRowAnimation:UITableViewRowAnimationFade];
    }
}

- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender
{
    if ([[segue identifier] isEqualToString:@"UpdateContacts"]) {
        NSManagedObject *selectedDevice = [self.contactarray
objectAtIndex:[self.tableView indexPathForSelectedRow] row];
        MyDetailViewController *destViewController = segue.destinationViewController;
        destViewController.contactdb = selectedDevice;
    }
}

```

25) Ready to Test!

- a. Add Contacts by clicking on + symbol
- b. Edit Contact by selecting row with data, make edits, save
- c. Delete Contact by Swipe Left on row and select Delete