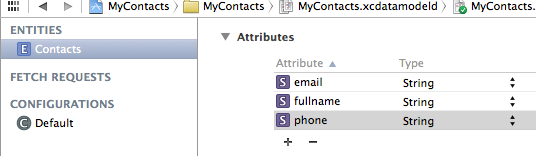
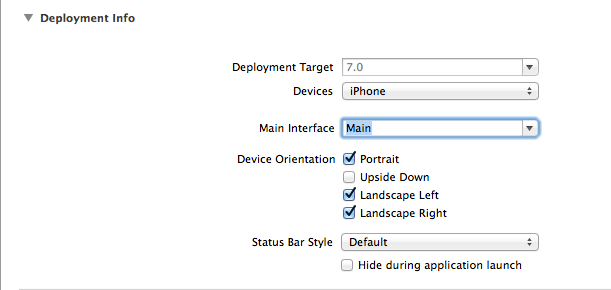
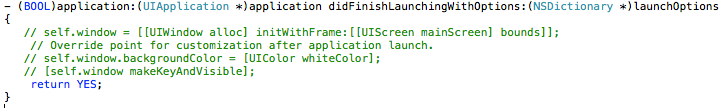
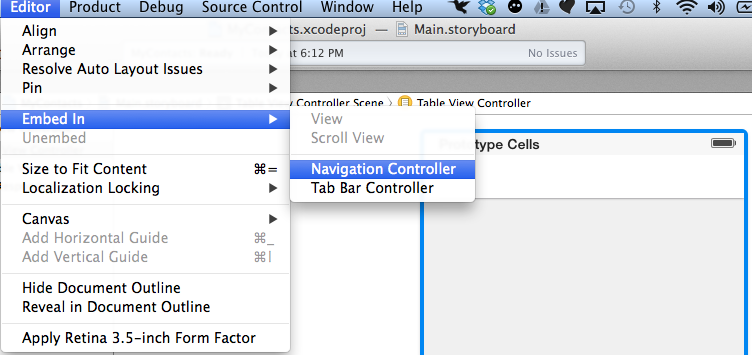
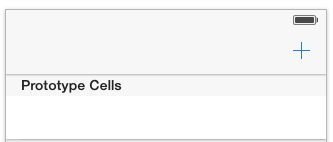
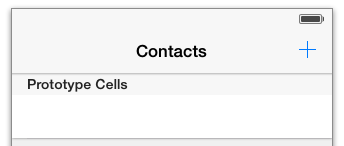
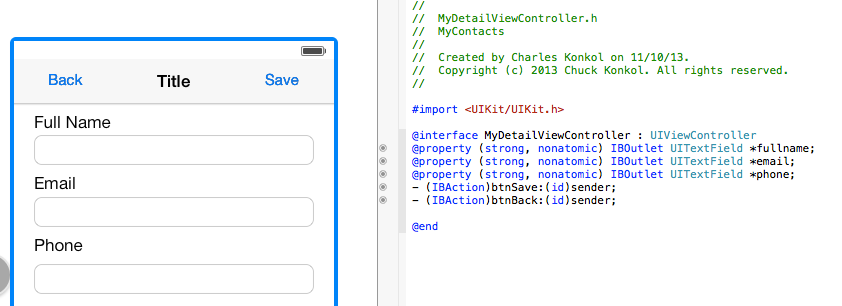
MyContacts database using CoreData. Insert, Update, and Delete

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
   1. Click the **+ Add Entity** button
   2. Double click **Entity** under **Entities** and rename to **Contacts**
4. Add Attributes to Entity **Contacts**
   1. Click the + symbol under **Attributes** and add the following attribute, type  
        
      **Attribute, Type**
   2. fullname, string
   3. email, string
   4. phone, string  
        
      
5. Add StoryBoard
   1. Right-click on MyContacts project name
   2. Select **New File** > **User Interface**
   3. Select **Storyboard**
   4. Save as **Main**
   5. Select **Create**
6. Select Main Interface
   1. Select **MyContacts** under project navigator
   2. Select under **Deployment Info** the dropdown **Main Interface** and select **Main.storyboard  
        
      **
7. Select **MyContactsAppDelegate.m** and edit **launchOptions**  
     
   
8. Add TableViewController
   1. Select **Main.storyboard**
   2. Add TableViewController
   3. Select **TableViewController**
   4. Embed in **Navigation Controller**
      1. **Editor > Embed In > Navigation Controller**
      2. **Save**
   5. Save  
        
      
9. Add **New File** > **Objective-C Class > Next**
   1. Name Class: **MyContactsViewController**
   2. Select Subclass: **UITableViewController**
   3. Select **Next**
   4. Select **Create**
10. Add class to MyContactsViewController
    1. Select **Main.storyboard**
    2. Select **TableViewController**
    3. Select **Identity Inspector**
       1. Select Dropdown
       2. Select class **MyContactsViewController**
       3. **Save**
11. Add new ViewController
    1. Select **Main.storyboard**
    2. Add **ViewController**
12. Add **New File** > **Objective-C Class > Next**
    1. Name Class: **MyDetailViewController**
    2. Select Subclass: **UIViewController**
    3. Select **Next**
    4. Select **Create**
13. Add class to **MyDetailViewController**
    1. Select **Main.storyboard**
    2. Select **ViewController**
    3. Select **Identity Inspector**
       1. Select Dropdown
       2. Select class **MyDetailViewController**
       3. **Save**
14. Add Button to right side of Navigation item bar
    1. Select Bar Button item from outline view
    2. Select Attribute **Type**
       1. Style: **Bordered**
       2. Identifier: **Add  
          **
15. Add Navigation Bar to **MyDetailViewController**  under battery icon
16. Add 2 buttonsNavigation Bar
    1. Left button title **Back**
       1. Style: **Bordered**
       2. Identifier: **Add**
    2. Right button title **Save**
       1. Style: **Bordered**
       2. Identifier: **Add**
17. Select MyContactsViewController from StoryBoard and set Title ofNavigation item bar **Contacts  
      
    **
18. Add controls to **MyDetailViewController** (make look like below)
    1. **3 Labels**
    2. **3 TextFields**
    3. Add **outlets** and **actions**
19. Control drag + button on **MyDetailViewController** to **MyContactsViewController** and select Modal
20. Select **Table View Cell** from **MyContactsViewController**
    1. Style: **Right Detail**
    2. Identifier: **Cell**
    3. Click on the **Modal segue from Cell to My Detail**
    4. Change identifier to **UpdateContacts**
21. **Modify MyDetailViewController.h** Add before @end@property (strong) NSManagedObject \*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 performSelector:@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];

}

1. Modify **MyDetailViewController.h** Add after @interface

@property (strong) NSMutableArray \*contactarray;

1. Modify **MyDetailViewController.m**
   1. #import "MyDetailViewController.h"
   2. Replace code below @implementation MyContactsViewController to @end

- (NSManagedObjectContext \*)managedObjectContext

{

NSManagedObjectContext \*context = nil;

id delegate = [[UIApplication sharedApplication] delegate];

if ([delegate performSelector:@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:@"%@ %@", [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! %@ %@", 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;

}

}

1. Ready to Test!
   1. Add Contacts by clicking on + symbol
   2. Edit Contact by selecting row with data, make edits, save
   3. Delete Contact by Swipe Left on row and select Delete