**Create a Storyboard that illustrates how to pass data using a custom class and the prepareForSegue method. We will Use Simon Allardice’s photo application from Lynda.com as a reference so you can retrace your steps at your convenience**

**These will be our steps**

1. Create Custom Class
2. Add Custom View Controller Classes
3. Load The Data
4. Pass the Object Between the Scenes

We will go through the finished version first.

**Part 1.**

**Create your project. Import photos folder…make sure you copy to the folder**

**Delete the Controllers on the storyboard**

**Delete Both Initial View Controller and h and m files**

**Replace with a table view controller**

**Embed the selected tableviewcontroller in Navigation Controller (Editor->Embed)**

**Add two view controllers to the story board**

**Hold control key/tap blank cell and drag to connect table view segue with second view controller**

**In ViewController 2:**

**Add image view**

**Add button**

**Create modal segue to ViewController3 by control - dragging from the button to the to 3rd view controller**

**Change background to 3rd view controller so that our text shows….maybe grayish…**

**Add Label – Allow Multiline….by selecting the lines in the attribute inspector to 0**

**Add finished button to the 3rd controller**

**Part 3: Creating the Custom Classes**

**Create custom class for code/overall functionality. Call it Photo and make it inherit from NSObject. This will have the general properties that will be inherited throughout our project**

**Photo.h:**

#import <Foundation/Foundation.h>

@interface Photo : NSObject

@property (nonatomic,strong) NSString \*name;

@property (nonatomic,strong) NSString \*filename;

@property (nonatomic,strong) NSString \*notes;

@end

**Add synthesizers to Photo.m**

#import "Photo.h"

@implementation Photo

@synthesize name;

@synthesize filename;

@synthesize notes;

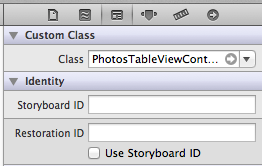
@end

**Create custom classes that inherit from their parent objects…**

**I called mine TableViewController (inherits from UITableViewController), DisplayViewController (inherits from UIViewController), and InfoViewController (inherits from UIViewController).**

**Do not check add xib as we already have interfaces on the storyboard!**

**Connect each xib with their class by using the Custom Class after you’ve selected each xib by the view controller button on the bottom of the screen so that a blue bar appears around the controller!**



**Import Photo.h to each of the interface classes #import “Photo.h”**

**Make sure each interface (h file) imports the class of the interface it is sending data to!!!!!**

**Add an NSMutableArray to the TableViewController.m (or whatever you named the**

NSMutableArray \*photos;

**It is not a property in the h file because no other class uses it.**

**Add a property in the Display Picture and Information interface files as they will need to display only one piece of info (Pic or information):**

@property (strong, nonatomic) Photo \*currentPhoto;

**Synthesize those properties in their respective implementation files (.m)!**

**Add properties for image view in the DisplayViewController and the label in the InfoViewController by making quick connections!**

**Part 3: Loading Data**

**Time to populate the data in the tables…**

**In our photosTableViewController.m, add the following to the viewDidLoad method.**

photos = [[NSMutableArray alloc] init];

Photo \*pic = [[Photo alloc] init];

[pic setName:@"Overlook"];

[pic setFilename:@"overlook.png"];

[pic setNotes:@"Looking out over the Pacific Ocean"];

[photos addObject:pic];

pic = [[Photo alloc] init];

[pic setName:@"Flag"];

[pic setFilename:@"flag.png"];

[pic setNotes:@"California flag, blowing in the wind against a blue sky."];

[photos addObject:pic];

pic = [[Photo alloc] init];

[pic setName:@"Beach"];

[pic setFilename:@"beach.png"];

[pic setNotes:@"Waves lapping against a California beach."];

[photos addObject:pic];

pic = [[Photo alloc] init];

[pic setName:@"Olives"];

[pic setFilename:@"olives.png"];

[pic setNotes:@"Olives ripening on a tree."];

[photos addObject:pic];

pic = [[Photo alloc] init];

[pic setName:@"Winery"];

[pic setFilename:@"winery.png"];

[pic setNotes:@"A sign for the Old Creek Ranch and Winery."];

[photos addObject:pic];

**Use jumpbar in PhotosTableViewController m file to display methods**

**These methods will be prepopulated because it is a tableviewcontroller!**

- (NSInteger)numberOfSectionsInTableView:(UITableView \*)tableView

{

// Return the number of sections.

**return 1;**

}

- (NSInteger)tableView:(UITableView \*)tableView numberOfRowsInSection:(NSInteger)section

{

// Return the number of rows in the section.

**return [photos count];**

}

Then….in the populate each cell method **(- (UITableViewCell \*)tableView:(UITableView \*)tableView cellForRowAtIndexPath:(NSIndexPath \*)indexPath):**

Photo \*current = [photos objectAtIndex:indexPath.row];

[cell.textLabel setText:[current name]];

**Now back to the storyboard….we need to give our 1 table cell an identifier…so highlight that cell and go to the attributes inspector….call it PhotoCell in the identifier area…make sure it matches the code in the table implementation file.**

**Run it!**

**Part 4 Passing Objects**

**Time to pass the info and objects from the selected table cell….**

**Name your segues by selecting them in the storyboard and giving them an identifier in the attributes inspector…for example the segue from table view to the display photo view could be named “ShowPhoto”**

**We’ll use prepareForSegue instead of the didSelectRowAtIndexPath**

**Add the following method to your tableview implementation (PhotosTableViewController.m) file:**

-(void) prepareForSegue:(UIStoryboardSegue \*)segue sender:(id)sender {

**//this condition tells us which segue has been chosen and is great for multiple segues….**

if ([segue.identifier isEqualToString:@"ShowPhoto"] ) {

**//this tells us where we will be segueing to....remember the DisplayViewController will display the photo.**

DisplayViewController \*dvc = [segue destinationViewController];

**//this tells us the table cell that we selected....**

NSIndexPath \*path = [self.tableView indexPathForSelectedRow];

**//this gets the current photo from the array**

Photo \*c = [photos objectAtIndex:[path row]];

[dvc setCurrentPhoto:c] ; }

**Now we need to go to the DisplayView Controller implementation file…(.m) In here, we will go to the ViewDidLoad method and set up our picture…**

**Paste the following into the viewDidLoad method:**

**//This will instantiate a new image object that will be equal to the filename //object for the selected table cell**

UIImage \*image = [UIImage imageNamed:[currentPhoto filename]];

**//This will set that image**

[self.currentImage setImage:image];

**//This will set that title**

[self setTitle:[currentPhoto name]];

**Run it…you should see some pics!**

**Now to prepareForSegue in the DisplayViewController to go to the InfoViewController**

**Add the following method to the DisplayViewController implementation file:**

-(void) prepareForSegue:(UIStoryboardSegue \*)segue sender:(id)sender {

**//this tells us where we are going**

InfoViewController \*ivc = [segue destinationViewController];

**//this will tell the information controller what photo we have selected**

[ivc setCurrentPhoto:[self currentPhoto]];

}

**//we do not need a segue name here because the photo chosen will dictate our information.**

**Now for our infoViewController to display the details….**

**//paste this into your viewDidLoad method in the information file…**

//**This simply sets the text of the label outlet with the value of the notes of the current object**

[detailsLabel setText:[currentPhoto notes]];

**Run it! You should see the details..**

**Time to add functionality to our last button….**

**Add an action to that button**

**Add this:** [self dismissViewControllerAnimated:NO completion:nil];

**\*note: [self dismissModalViewControllerAnimated:YES]; is deprecated…**

**To your newly created method….**

**Last thing: Change the last segue’s transition to partial curl**

**Try some other transisionts**

**Run it…**