We Will Review Our StoryBoard Segue Application From Last Week and Build Something new so we can pass data from Scene to scene

**Step 1:** Create a new project using the Single View Application template. On the project options panel enter Storyboard as both the product name and class prefix.

**Step 2:** Drag a label and a button onto the view canvas. Using the properties panel, change the label text to Scene One and the button text to Go to Scene 2.

**Step 3:** Create an outlet for that label named sceneLabel1

@property (strong, nonatomic) IBOutlet UILabel \*scene1Label;

**Step 4:** To add a second scene to the storyboard drag a View Controller object from the Object Library panel onto the canvas

**Step 5:** Drag and drop a label and a button into the second scene. Have the text for the label be “scene 2” and the text for the button be “back to scene 1”

**Step 6:** Create a segue from the button in scene1 to the scene 2. Make the transition a modal transition

**Step 7:** Associate that second view with a class**.** Create a new objective-c class from the cocoa touch library and name it Scene2ViewController. Make sure this new class inherits or is a subclass of the UIViewController Class. In the identity inspector, associate that new view with that new class.

**Step 8:**  Now with the new class, we can hook up the elements for the second controller, so select the label object in scene 2 and display the Assistant Editor. Make sure that the Scene2ViewController.h file is displayed in the editor and then establish an outlet for the label named scene2Label.

**NOW FOR PASSING DATA**

One of the most common requirements when working with storyboards involves passing data from one scene to another during a segue transition. This is achieved using the prepareForSegue method.

Before a segue is performed by the storyboard runtime environment, a call is made to the prepareForSegue method of the current view controller. If any tasks need to be performed prior to the segue taking place, implement this method in the current view controller and add code to perform any necessary tasks. Passed as an argument to this method is a segue object from which a reference to the destination view controller may be obtained and subsequently used to transfer data.

**Step 9:** Add some new properties to the Scene2ViewController.h

**@property (strong, nonatomic) IBOutlet UILabel \*scene2Label;**

**@property (strong, nonatomic) NSString \*labelText**

**Step 10:** Now add the following to the ViewDidLoad Method in the Scene2ViewController.m

**\_scene2Label.text = \_labelText;**

**Step 11:** Go back to the first view controller’s h file (StoryBoardViewController.h) and import the second controller’s interface file.

**#import "Scene2ViewController.h"**

This is so we can have access to the second controller’s properties in the first controller.

**Step 12:** Now in the StoryBoardViewController.m, we can finally implement the segue.

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

**{**

**Scene2ViewController \*destination =**

**[segue destinationViewController];**

**destination.labelText = @"Arrived from Scene 1";**

**}**

**Step 13:** Unwind the segue from scene 2 using the button on scene 2. Add the following method to the StoryBoardViewController.m.

-(IBAction)returned:(UIStoryboardSegue \*)segue {

\_scene1Label.text = @"Returned from Scene 2";

}

Now Control click the button and drag to the Exit icon below the view.

You should see an unwind option pop up similar to last week. Select the returned method.

**NOTE: You can trigger a segue programmatically by giving it an identifier and the using the following code:**

**[self performSegueWithIdentifier: @"SegueToScene1"**

**sender: self];**