

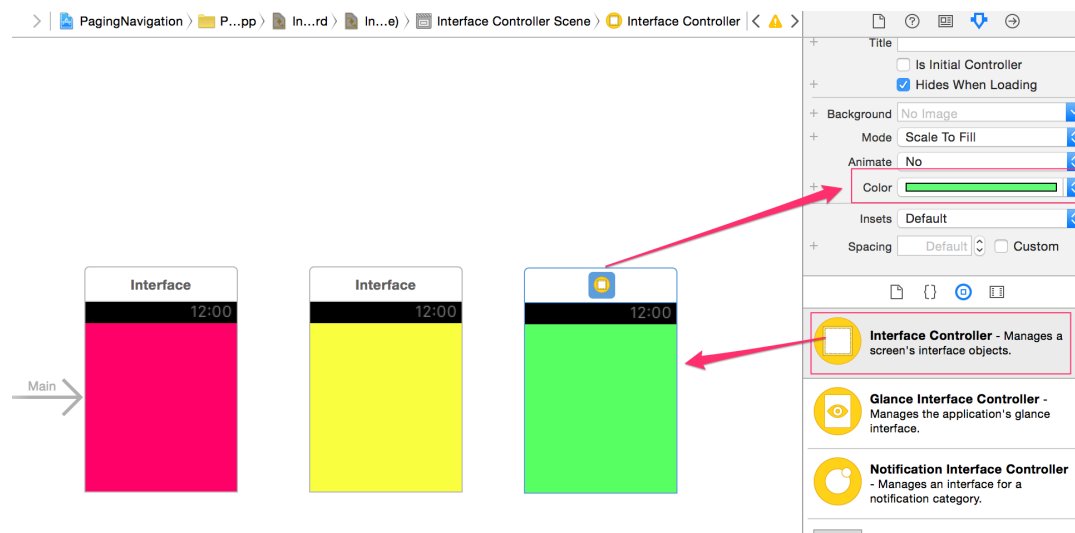
WK Paging Navigation Tutorial

Start a New WatchKit Project

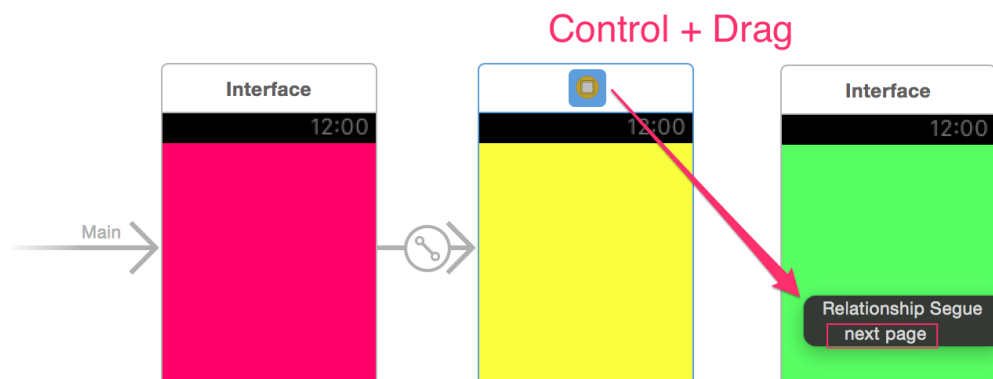
Follow the **Hello, WatchKit Tutorial** to set up a new WatchKit project. Make sure your Watch App runs properly.

Page-Based Navigation Setup

1. In your Watch App's **Interface.storyboard**, add a few **Interface Controllers** and change their background colors to easily identify them later on



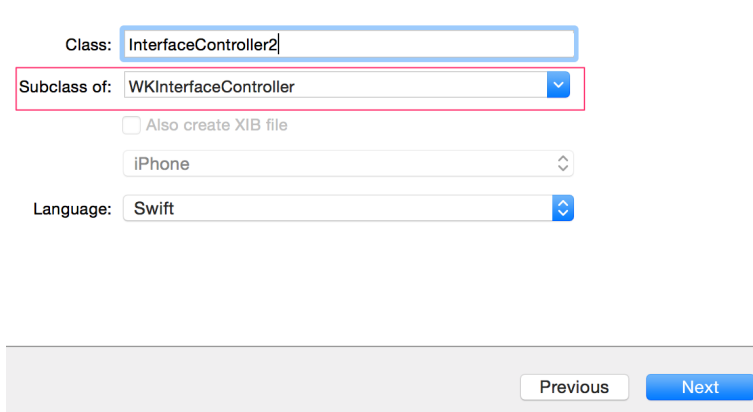
2. **Control + Drag** between the Interface Controller to create **next page Relationship Segues**.



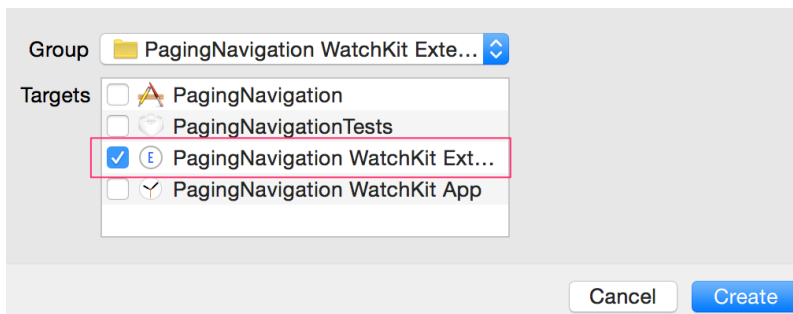
3. That's it! Run the app to see how it works!

Experiment With the Page-Based Controller Life Cycle

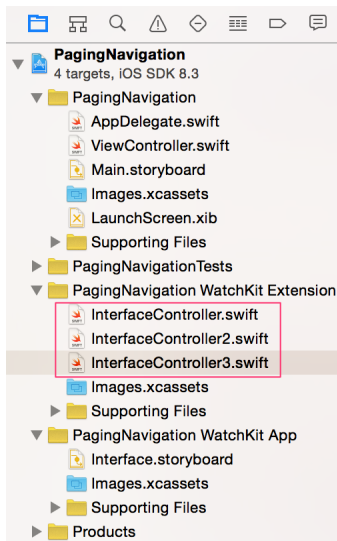
1. Create Interface Controllers for the other two controllers
 - In the **Xcode menu**, select **File > New > File**
 - Select the **Cocoa Touch Class** option and tap **Next**
 - Make your **InterfaceController2** (or your own name), a **subclass of WKInterfaceController** and tap **Next**



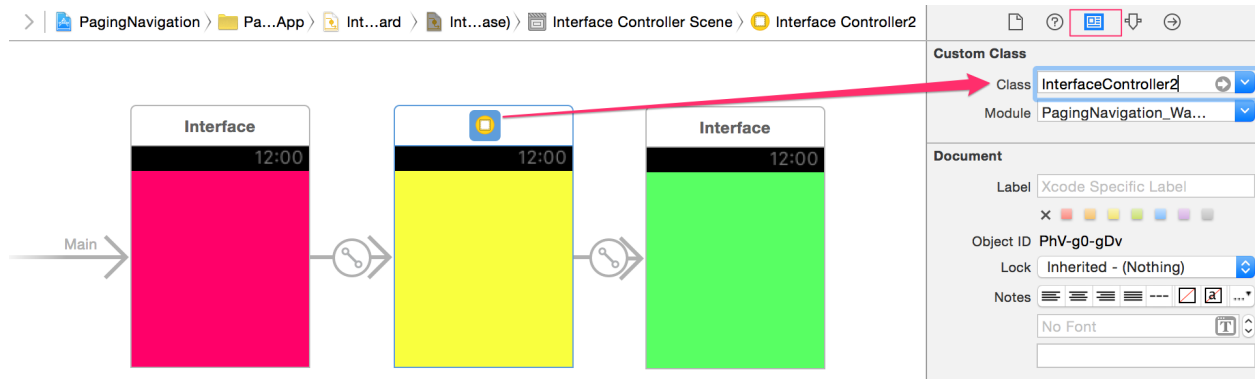
- Make sure to select the **WatchKit Extension Target** for your new Interface Controller and tap **Create**



- Repeat for your other Interface Controllers



2. Set the **Custom Class** for your newly created Interface Controllers in the Storyboard



3. Add print statements in the life cycle methods for each of your Interface Controllers (make sure to change out the name of the Interface Controller so you can tell them apart when they print!)

```
class InterfaceController2: WKInterfaceController {  
  
    override func awakeWithContext(context: AnyObject?) {  
        super.awakeWithContext(context)  
        println("awakeWithContext: InterfaceController2")  
    }  
  
    override func willActivate() {  
        super.willActivate()  
        println("willActivate: InterfaceController2")  
    }  
  
    override func didDeactivate() {  
        super.didDeactivate()  
        println("didDeactivate: InterfaceController2")  
    }  
}
```

4. Run your app and monitor the print statements as you navigate between pages to understand the page-based navigation controller life cycles.