

## Beginning iPhone SDK Programming with Objective-C

By Wei-Meng Lee

### Notes on Changes in Interface Builder (3.2.1)

As much as I strive to keep my book as up-to-date as possible, it is always not possible to do so, especially when you are writing a book on the iPhone SDK.

During the course of writing this book, I was using version 3.0 of the SDK. In version 3.0 of the iPhone SDK, Interface Builder allows you to create *outlets* and *actions* in the **Identity Inspector** window, as shown in the various figures in the book. An example is the [Figure 2-21](#), shown on Page 36.

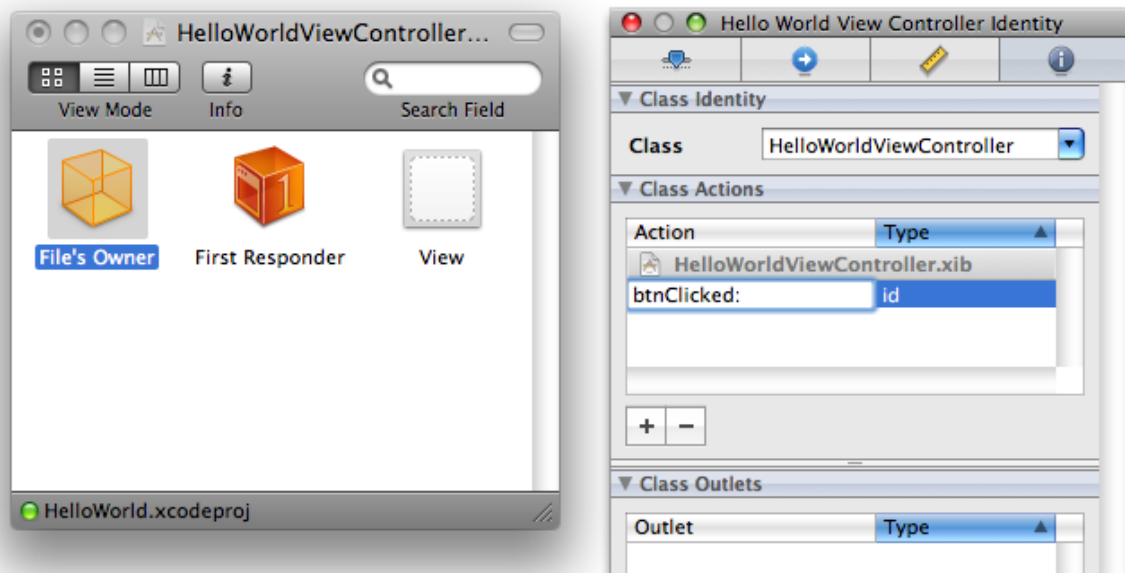
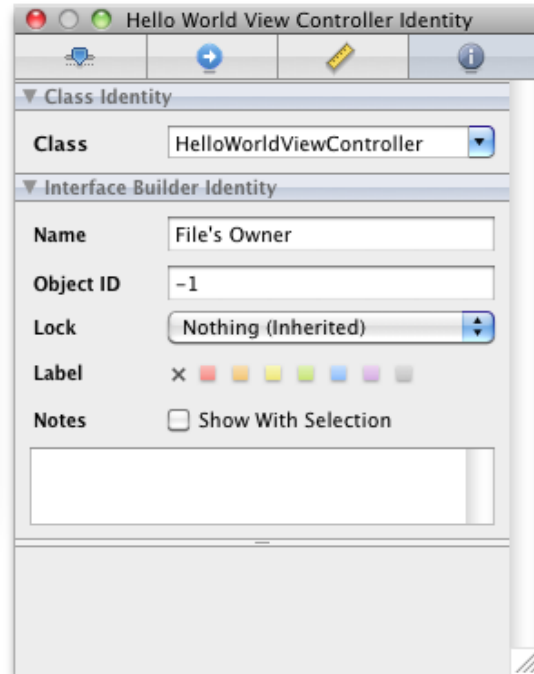
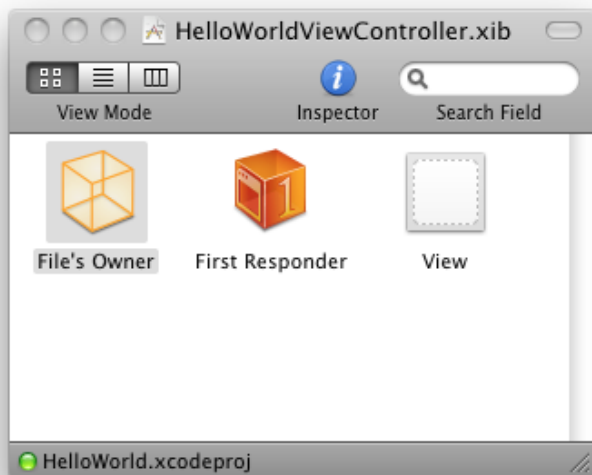


Figure 2-21

However, in the latest version of the iPhone SDK (3.1.2), Interface Builder (3.2.1) removed the **Class Actions** and **Class Outlets** sections in the **Identity Inspector** window:



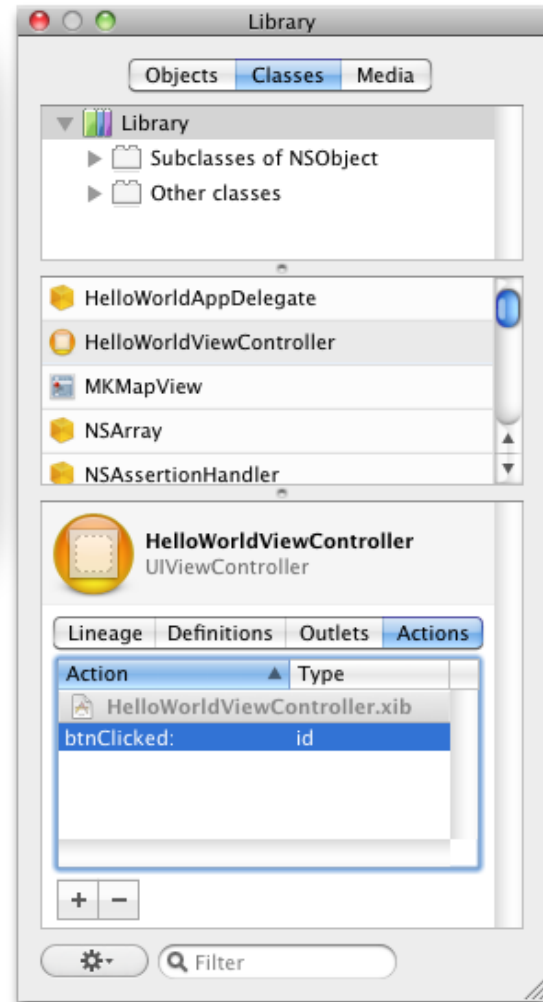
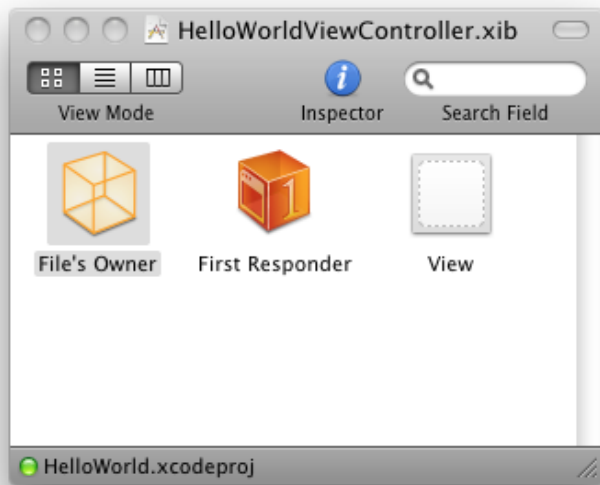
This poses a little problem for readers of my book, who are expecting to create outlets and actions in the **Identity Inspector** window. There are two easy ways to solve the problem, and I am going to detail them below.

### Method 1

Instead of creating outlets and actions in the **Identity Inspector** window, simply declare them in the header (.h) file. Once you have declared the outlets and actions in the .h file, you can go back to Interface Builder and perform the connections. In my book, I often show creating the outlets and actions in Interface Builder first, and then declare them in the .h file. So, with the latest version of Interface Builder, you just need to skip the first step and jump directly to the second step (declare them in the .h file).

### Method 2

If you still want to create outlets and actions in Interface Builder, you now have to go to the **Library** window (**Tools**→**Library**). Click on the **Classes** tab (see figure below), select the view controller (e.g. **HelloWorldViewController**) and then create the outlets or actions below:



Once the outlets and actions are created, you would need to declare them in our .h file, as usual.