



iOS Development

iOS Development Tips and Tricks

Method Documentation

// MARK: - Table view delegate

```
C @interface ProfileTableViewController()
C @implementation ProfileTableViewController
P @synthesize profilesArray, fugitiveViewController;
M -initWithStyle:
M -viewDidLoad
M -viewWillAppear:
M -viewDidUnload
M -shouldAutorotateToInterfaceOrientation:
```

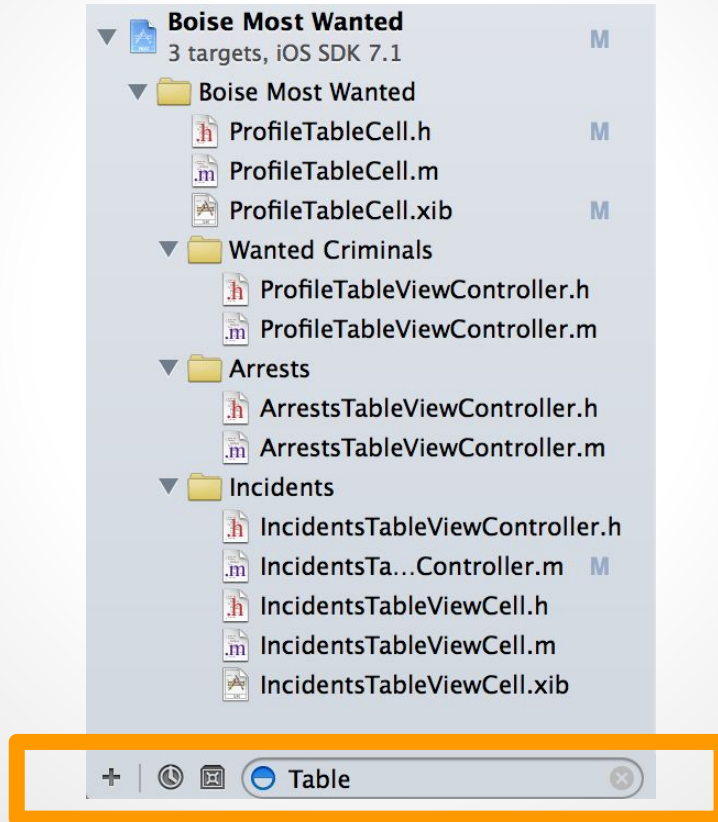
Table view data source

```
M -numberOfSectionsInTableView:
M -tableView:numberOfRowsInSection:
M -tableView:cellForRowAtIndexPath:
```

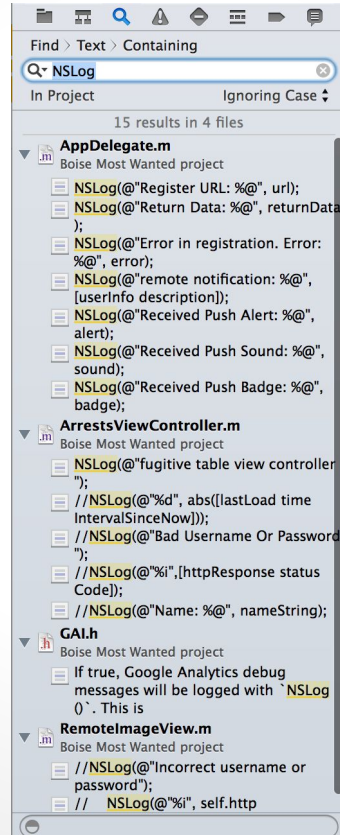
Table view delegate

```
M -tableView:heightForRowAtIndexPath:
M -tableView:didSelectRowAtIndexPath:
```

File Name Search



Search Within Files Command-4



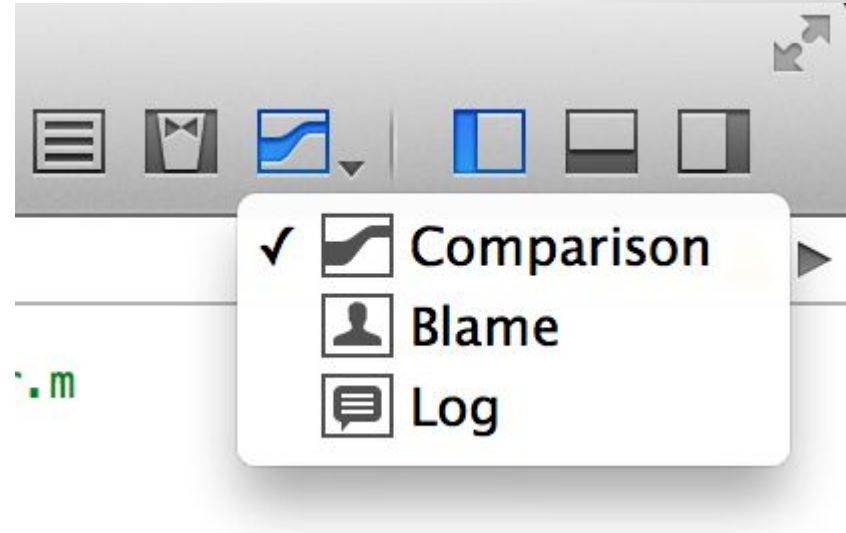
Source Control

Comparison -

Compare revisions

Blame - Find out who's to blame

Log - See the revision comments



Refactor - Rename



Refactor - Extract

Extract to

☒ New method ☐ New function

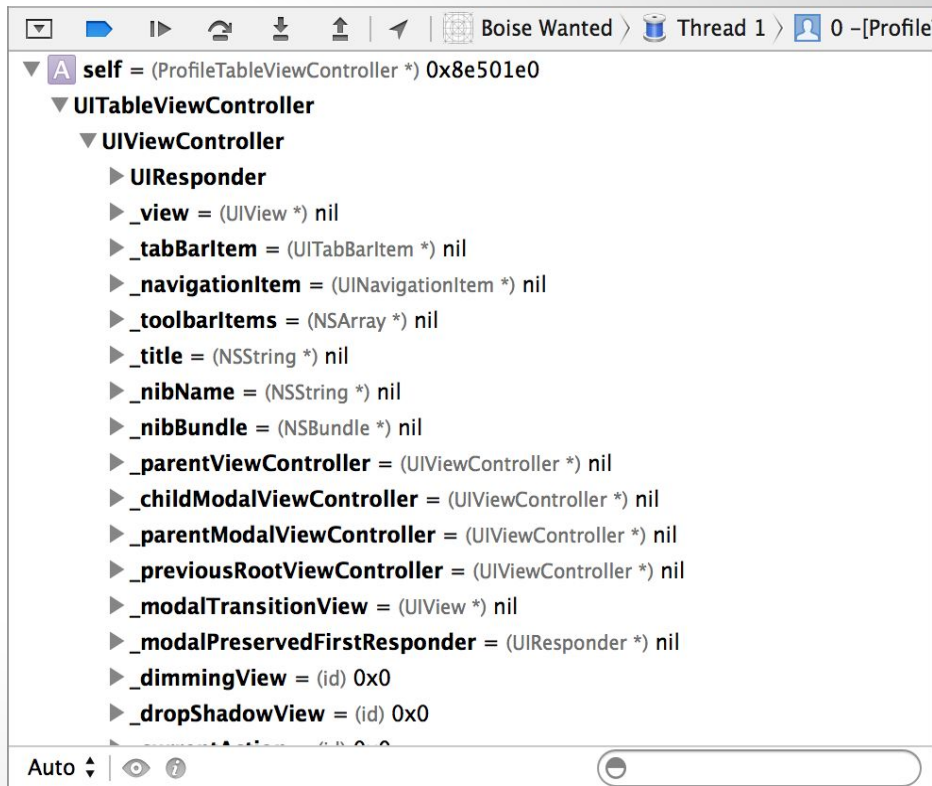
25	- (void)extracted_method	1 ▾	25
26	{		26
27	CGRect screenFrame = [[UIScreen mainScreen] bounds];		27
28	UILabel *centeredLabel = [[UILabel alloc] initWithFrame: CGRectMake(0,	2 ▾	28
29	centeredLabel.text = @"Label Text";		29
30	centeredLabel.backgroundColor = [UIColor clearColor];		30
31	centeredLabel.font = [UIFont fontWithName:@"Helvetica" size: 14];		31
32	[self.window addSubview: centeredLabel];		32

Debug Area

Filter Variables

Quick Look

Print Description

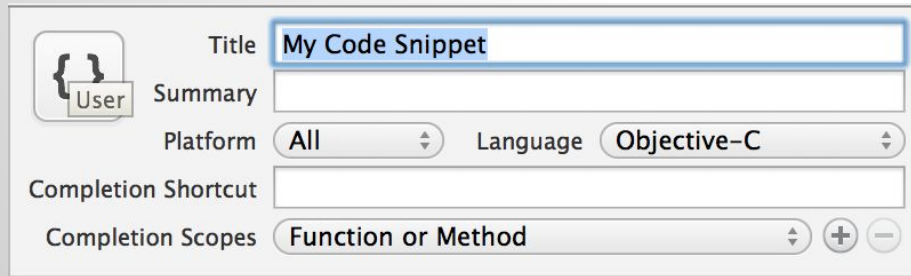


Debug Area

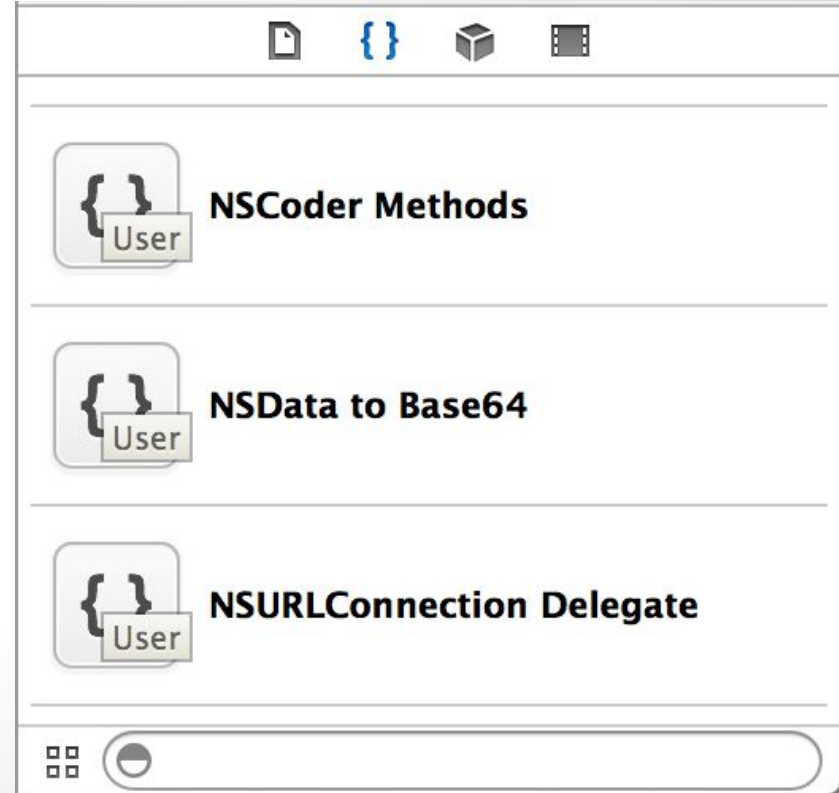
Demo

Code Snippets

Add your own,
commonly used snippets
of code.



A screenshot of the Code Snippet editor interface. It features a sidebar on the left with a 'User' icon. The main area contains a form with the following fields: 'Title' (containing 'My Code Snippet'), 'Summary' (empty), 'Platform' (set to 'All'), 'Language' (set to 'Objective-C'), 'Completion Shortcut' (empty), and 'Completion Scopes' (set to 'Function or Method').



Quick Help - Option-Click

```
static var persistentContainer: NSPersistentContainer = {
```

```
/*
```

```
The persistentContainer property  
creates and manages the persistent  
application container. If the  
error condition is not nil, the  
error condition is nil.
```

```
*/
```

```
let container = NSPersistentContainer(name: "MyApp")  
container.loadPersistentStores(completionHandler: { (storeDescription, error) in
```

```
    if let error = error {  
        fatalError("Unrecognized error: \(error)")  
    }  
    // For more information about this implementation, see the documentation for the  
    // fatalError() causes the application to generate a crash log and terminate the app.  
    // This function is called from the UIApplication class, although it may be useful in other contexts.
```

Declaration `class NSPersistentContainer : NSObject`

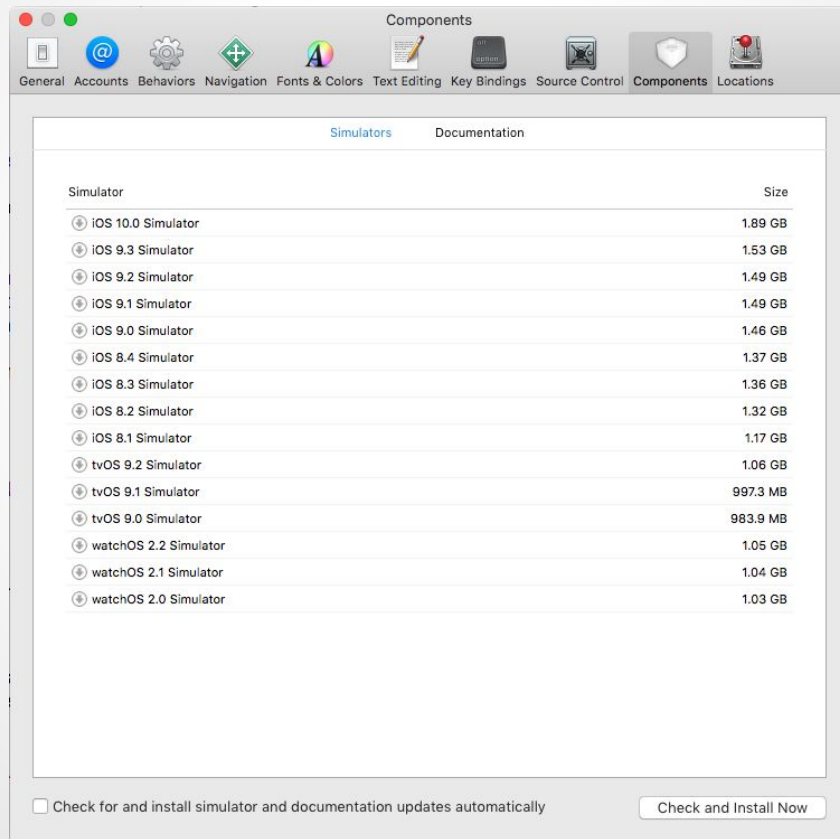
Description A container that encapsulates the Core Data stack in your application. NSPersistentContainer simplifies the creation and management of the Core Data stack by handling the creation of the `NSManagedObjectModel`, `NSPersistentStoreCoordinator`, and the `NSManagedObjectContext`.

Availability iOS (10.0 and later), macOS (10.12 and later), tvOS (10.0 and later), watchOS (3.0 and later)

Declared In `Core Data`

More [Class Reference](#)

Adding Older Simulators - CMD-,



Compiler Macros and Directives

Useful for including or excluding code during compilation

▼ Apple LLVM 5.1 – Preprocessing	
Setting	
Enable Foundation Assertions	Yes ↕
▼ Preprocessor Macros	<Multiple values>
Debug	QA=1 DEBUG=1
Release	QA=1
Preprocessor Macros Not Used In Precompiled Headers	

Compiler Macros and Directives - Example

```
#ifdef PROD
```

```
let SERVER_LOCATION = "http://www.production.com"
```

```
#endif
```

```
#ifdef QA
```

```
let SERVER_LOCATION = "http://test.production.com/"
```

```
#endif
```

```
#ifdef DEV
```

```
let SERVER_LOCATION = "http://test.zstudiolabs.com"
```

```
#endif
```