
UITableViewController Class Reference

User Experience: Tables



2010-02-25



Apple Inc.
© 2010 Apple Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Bonjour, iPhone, and Objective-C are trademarks of Apple Inc., registered in the United States and other countries.

iOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE

ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

UITableViewController Class Reference 5

Overview 5

Tasks 6

 Initializing the UITableViewController Object 6

 Getting the Table View 6

 Configuring the Table Behavior 6

Properties 6

 clearsSelectionOnViewWillAppear 6

 tableView 7

Instance Methods 7

 initWithStyle: 7

Document Revision History 9

UITableViewController Class Reference

Inherits from	UIViewController : UIResponder : NSObject
Conforms to	UITableViewDelegate UITableViewDataSource NSCoding (UIViewController) NSObject (NSObject)
Framework	/System/Library/Frameworks/UIKit.framework
Availability	Available in iOS 2.0 and later.
Companion guide	Table View Programming Guide for iOS
Declared in	UITableViewController.h
Related sample code	BonjourWeb GKRocket MultipleDetailViews ToolbarSearch WiTap

Overview

The `UITableViewController` class creates a controller object that manages a table view. It implements the following behavior:

- If a nib file is specified via the `initWithNibName:bundle:` method (which is declared by the superclass `UIViewController`), `UITableViewController` loads the table view archived in the nib file. Otherwise, it creates an unconfigured `UITableView` object with the correct dimensions and autoresize mask. You can access this view through the `tableView` (page 7) property.
- If a nib file containing the table view is loaded, the data source and delegate become those objects defined in the nib file (if any). If no nib file is specified or if the nib file defines no data source or delegate, `UITableViewController` sets the data source and the delegate of the table view to `self`.
- When the table view is about to appear the first time it's loaded, the table-view controller reloads the table view's data. It also clears its selection (with or without animation, depending on the request) every time the table view is displayed. The `UITableViewController` class implements this in the superclass method `viewWillAppear:`. You can disable this behavior by changing the value in the `clearsSelectionOnWillAppear` (page 6) property.
- When the table view has appeared, the controller flashes the table view's scroll indicators. The `UITableViewController` class implements this in the superclass method `viewDidAppear:`.

- It implements the superclass method `setEditing:animated:` so that if a user taps an Edit|Done button in the navigation bar, the controller toggles the edit mode of the table.

You create a custom subclass of `UITableViewController` for each table view that you want to manage. When you initialize the controller in `initWithStyle:` (page 7), you must specify the style of the table view (plain or grouped) that the controller is to manage. Because the initially created table view is without table dimensions (that is, number of sections and number of rows per section) or content, the table view's data source and delegate—that is, the `UITableViewController` object itself—must provide the table dimensions, the cell content, and any desired configurations (as usual). You may override `loadView` or any other superclass method, but if you do be sure to invoke the superclass implementation of the method, usually as the first method call.

Tasks

Initializing the UITableViewController Object

- `initWithStyle:` (page 7)

Initializes a table-view controller to manage a table view of a given style.

Getting the Table View

`tableView` (page 7) *property*

Returns the table view managed by the controller object.

Configuring the Table Behavior

`clearsSelectionOnViewWillAppear` (page 6) *property*

A Boolean value indicating if the controller clears the selection when the table appears.

Properties

For more about Objective-C properties, see “Properties” in *The Objective-C Programming Language*.

`clearsSelectionOnViewWillAppear`

A Boolean value indicating if the controller clears the selection when the table appears.

```
@property(nonatomic) BOOL clearsSelectionOnViewWillAppear
```

Discussion

The default value of this property is YES. When YES, the table view controller clears the table's current selection when it receives a `viewWillAppear:` message. Setting this property to NO preserves the selection.

Availability

Available in iOS 3.2 and later.

Declared In

UITableViewController.h

tableView

Returns the table view managed by the controller object.

```
@property(nonatomic, retain) UITableView *tableView
```

Availability

Available in iOS 2.0 and later.

Related Sample Code

GKRocket

MultipleDetailViews

ToolbarSearch

Declared In

UITableViewController.h

Instance Methods

initWithStyle:

Initializes a table-view controller to manage a table view of a given style.

```
- (id)initWithStyle:(UITableViewStyle)style
```

Parameters

style

A constant that specifies the style of table view that the controller object is to manage (UITableViewStylePlain or UITableViewStyleGrouped).

Return Value

An initialized UITableViewController object or nil if the object couldn't be created.

Discussion

If you use the standard `init` method to initialize a UITableViewController object, a table view in the plain style is created.

Availability

Available in iOS 2.0 and later.

Related Sample Code

ToolbarSearch

Declared In

UITableViewController.h

Document Revision History

This table describes the changes to *UITableViewController Class Reference*.

Date	Notes
2010-02-25	Updated for iOS 3.2.
2009-03-26	Clarified when class calls reloadData.
2008-06-04	New document that describes the class for creating a controller object that manages a table view.

REVISION HISTORY

Document Revision History