
UILocalizedIndexedCollation Class Reference

Data Management



2010-05-11



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, 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

UILocalizedIndexedCollation Class Reference 7

Overview 7

Tasks 8

 Getting the Shared Instance 8

 Preparing the for Sections and Section Indexes 8

 Providing Section Index Data to the Table View 8

Properties 8

 sectionIndexTitles 8

 sectionTitles 9

Class Methods 9

 currentCollation 9

Instance Methods 10

 sectionForObject:collationStringSelector: 10

 sectionForSectionIndexTitleAtIndex: 10

 sortedArrayFromArray:collationStringSelector: 11

Document Revision History 13

Listings

[UILocalizedIndexedCollation Class Reference](#) 7

[Listing 1](#) [Data source using indexed-collation object to provide data to table view](#) 7

UILocalizedIndexedCollation Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/UIKit.framework
Availability	Available in iOS 3.0 and later.
Declared in	UILocalizedIndexedCollation.h

Overview

The `UILocalizedIndexedCollation` class is a convenience for organizing, sorting, and localizing the data for a table view that has a section index. The table view's data source then uses the collation object to provide the table view with input for section titles and section index titles.

Table views with section indexes are ideal for displaying and facilitating the access of data composed of many items organized by a sequential ordering scheme such as the alphabet. Users tap an index title to jump to the corresponding section. The initial table view of the Phone/Contacts application on the iPhone is an example. Note that the section titles can be different than the titles of the index.

To prepare the data for a section index, the `UITableViewController` object creates a indexed-collation object and then, for each model object that is to be indexed, calls `sectionForObject:collationStringSelector:` (page 10). This method determines the section in which each of these objects should appear and returns an integer that identifies the section. The table-view controller then puts each object in a local array for its section. For each section array, the controller calls the `sortedArrayFromArray:collationStringSelector:` (page 11) method to sort all of the objects in the section. The indexed-collation object is now the data store that the table-view controller uses to provide section-index data to the table view, as illustrated in Listing 1.

Listing 1 Data source using indexed-collation object to provide data to table view

```
- (NSString *)tableView:(UITableView *)tableView
titleForHeaderInSection:(NSInteger)section
{
    return [[[UILocalizedIndexedCollation currentCollation] sectionTitles]
objectAtIndex:section];
}

- (NSArray *)sectionIndexTitlesForTableView:(UITableView *)tableView
{
    return [[[UILocalizedIndexedCollation currentCollation] sectionIndexTitles];
}
```

```

- (NSInteger)tableView:(UITableView *)tableView
sectionForSectionIndexTitle:(NSString *)title atIndex:(NSInteger)index
{
    return [[UILocalizedIndexedCollation currentCollation]
sectionForSectionIndexTitleAtIndex:index];
}

```

Tasks

Getting the Shared Instance

- + [currentCollation](#) (page 9)
Returns the shared indexed-collation instance.

Preparing the for Sections and Section Indexes

- [sectionForObject:collationStringSelector:](#) (page 10)
Returns an integer identifying the section in which a model object belongs.
- [sortedArrayFromArray:collationStringSelector:](#) (page 11)
Sorts the objects within a section by their localized titles.

Providing Section Index Data to the Table View

- [sectionTitles](#) (page 9) *property*
Returns the list of section titles for the table view. (read-only)
- [sectionIndexTitles](#) (page 8) *property*
Returns the list of section-index titles for the table view (read-only)
- [sectionForSectionIndexTitleAtIndex:](#) (page 10)
Returns the section that the table view should scroll to for the given index title.

Properties

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

sectionIndexTitles

Returns the list of section-index titles for the table view (read-only)


```
@property(nonatomic, readonly) NSArray *sectionIndexTitles
```

Discussion

This property contains the localized list of section-index titles sorted according to the specified ordering (for example, A through Z in US English). In its implementation of `sectionIndexTitlesForTableView:`, the data source can call this method on the indexed-collation object and pass back the result.

Availability

Available in iOS 3.0 and later.

See Also

[@property sectionTitles](#) (page 9)
[- sectionForSectionIndexTitleAtIndex:](#) (page 10)

Declared In

UILocalizedIndexedCollation.h

sectionTitles

Returns the list of section titles for the table view. (read-only)

```
@property(nonatomic, readonly) NSArray *sectionTitles
```

Discussion

This property contains the localized list of section titles sorted according to the specified ordering (for example, A through Z in US English). In its implementation of `tableView:titleForHeaderInSection:`, the data source can call this method on the indexed-collation object, passing in the section index and returning the result.

Availability

Available in iOS 3.0 and later.

See Also

[@property sectionIndexTitles](#) (page 8)
[- sectionForSectionIndexTitleAtIndex:](#) (page 10)

Declared In

UILocalizedIndexedCollation.h

Class Methods

currentCollation

Returns the shared indexed-collation instance.

```
+ (id)currentCollation
```

Return Value

A `UILocalizedIndexedCollation` object or `nil` if there was a problem creating the object.

Availability

Available in iOS 3.0 and later.

Declared In

UILocalizedIndexedCollation.h

Instance Methods

sectionForObject:collationStringSelector:

Returns an integer identifying the section in which a model object belongs.

```
- (NSInteger)sectionForObject:(id)object collationStringSelector:(SEL)selector
```

Parameters

object

A model object of the application that is part of the data model for the table view.

selector

A selector that identifies a method returning an identifying string for *object* that is used in collation. The method should take no arguments and return an `NSString` object. For example, this could be a `name` property on the object.

Return Value

An integer that identifies the section in which the model object belongs. The numbers returned indicate a sequential ordering.

Discussion

The table-view controller should iterate through all model objects for the table view and call this method for each object. If the application provides a `Localizable.strings` file for the current language preference, the indexed-collation object localizes each string returned by the method identified by *selector*. It uses this localized name when collating titles. The controller should use the returned integer to identify a local “section” array in which it should insert *object*.

Availability

Available in iOS 3.0 and later.

See Also

- [sortedArrayFromArray:collationStringSelector:](#) (page 11)

Declared In

UILocalizedIndexedCollation.h

sectionForSectionIndexTitleAtIndex:

Returns the section that the table view should scroll to for the given index title.

```
- (NSInteger)sectionForSectionIndexTitleAtIndex:(NSInteger)indexTitleIndex
```

Parameters*indexTitleIndex*

An integer identifying a section-index title by its position in the array of such titles.

Return Value

An integer identifying the table-view section associated with *indexTitleIndex*.

Discussion

This method allows the table view to map between a given item in the section index and a given section even when there isn't a one-to-one mapping. In its implementation of `tableView:sectionForSectionIndexTitle:atIndex:`, the data source can call this method on the indexed-collation object specifying as an argument the passed-in index integer; it then returns the result to the table view.

Availability

Available in iOS 3.0 and later.

See Also

[@property sectionTitles](#) (page 9)

[@property sectionIndexTitles](#) (page 8)

Declared In

UILocalizedIndexedCollation.h

sortedArrayFromArray:collationStringSelector:

Sorts the objects within a section by their localized titles.

```
- (NSArray *)sortedArrayFromArray:(NSArray *)array
    collationStringSelector:(SEL)selector
```

Parameters*array*

An array containing the model objects for a section.

selector

A selector that identifies a method returning an identifying string for each object in *array*. The index-collation object uses this string for sorting the objects in the array. The method should take no arguments and return an `NSString` object. For example, this could be a `name` property on the object.

Return Value

A new array containing the items in *array*, sorted.

Discussion

The table-view controller creates the array of objects for a section (*array*) as part of iterating through its model objects with calls to the [sectionForObject:collationStringSelector:](#) (page 10) method. This method should be called on each local section array.

Availability

Available in iOS 3.0 and later.

Declared In

UILocalizedIndexedCollation.h

Document Revision History

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

Date	Notes
2010-05-11	Corrected sample code in class overview.
2009-05-19	Made minor corrections.
2009-03-08	First version of the document that describes the class that helps to organize data for table views with section indexes.

REVISION HISTORY

Document Revision History