

UILocalizedIndexedCollation

An object that provides ways to organize, sort, and localize the data for a table view that has a section index.

Declaration

```
class UILocalizedIndexedCollation : NSObject
```

Overview

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. The table view's data source uses the collation object to provide the table view with input for section titles and section index titles.

To prepare the data for a section index, your table-view controller creates an indexed-collation object and then, for each model object that is to be indexed, calls `section(for:collationStringSelector:)`. 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 `sortedArray(from:collationStringSelector:)` 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

Topics

Getting the Shared Instance

```
class func current() -> Self
```

Returns an indexed-collation instance for the current table view.

Preparing the Sections and Section Indexes

```
func section(for: Any, collationStringSelector: Selector) -> Int
```

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

```
func sortedArray(from: [Any], collationStringSelector: Selector) -> [Any]
```

Sorts the objects within a section by their localized titles.

Providing Section Index Data to the Table View

```
var sectionTitles: [String]
```

Returns the list of section titles for the table view.

```
var sectionIndexTitles: [String]
```

Returns the list of section-index titles for the table view

```
func section(forSectionIndexTitle: Int) -> Int
```

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