# NSSortDescriptor Class Reference

**Cocoa > Data Management**

**2007-10-31**

Apple Inc.
© 2007 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, Cocoa, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Spotlight is a trademark of Apple Inc.

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


## NSSortDescriptor Class Reference 5

Overview 5
Adopted Protocols 6
Tasks 6
- Initializing a Sort Descriptor 6
- Getting Information About a Sort Descriptor 6
- Using Sort Descriptors 6

Instance Methods 7
- ascending 7
- compareObject:toObject: 7
- initWithKey:ascending: 8
- initWithKey:ascending:selector: 8
- key 9
- reversedSortDescriptor 9
- selector 10

## Document Revision History 11

## Index 13




2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

# NSSortDescriptor Class Reference

| | |
|---|---|
| **Inherits from** | NSObject |
| **Conforms to** | NSCoding<br>NSCopying<br>NSObject (NSObject) |
| **Framework** | /System/Library/Frameworks/Foundation.framework |
| **Availability** | Available in Mac OS X v10.3 and later. |
| **Companion guide** | Sort Descriptor Programming Topics |
| **Declared in** | NSSortDescriptor.h |
| **Related sample code** | CoreRecipes<br>Departments and Employees<br>iSpend<br>NSOperationSample<br>SpotlightFortunes |

## Overview

An instance of `NSSortDescriptor` describes a basis for ordering objects by specifying the property to use to compare the objects, the method to use to compare the properties, and whether the comparison should be ascending or descending. Instances of `NSSortDescriptor` are immutable.

You construct an instance of `NSSortDescriptor` by specifying the key path of the property to be compared, the order of the sort (ascending or descending), and (optionally) a selector to use to perform the comparison. The three-argument constructor allows you to specify other comparison selectors such as `caseInsensitiveCompare:` and `localizedCompare:`. Sorting raises an exception if the objects to be sorted do not respond to the sort descriptor's comparison selector.

**Note:** Many of the descriptions of `NSSortDescriptor` methods refer to "property key". This, briefly, is a string (key) that identifies a property (an attribute or relationship) of an object. You can find a discussion of this terminology in "Object Modeling" in *Cocoa Fundamentals Guide* and in *Key-Value Coding Programming Guide*.

There are a number of situations in which you can use sort descriptors, for example:

- To sort an array (an instance of `NSArray` or `NSMutableArray`—see `sortedArrayUsingDescriptors:` and `sortUsingDescriptors:`)


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

- To directly compare two objects (see compareObject:toObject: (page 7))
- To specify how the elements in a table view should be arranged (see `sortDescriptors`)
- To specify how the elements managed by an array controller should be arranged (see `sortDescriptors`)
- If you are using Core Data, to specify the ordering of objects returned from a fetch request (see `sortDescriptors`)

# Adopted Protocols

NSCoding
- `encodeWithCoder:`
- `initWithCoder:`

NSCopying
- `copyWithZone:`

# Tasks

## Initializing a Sort Descriptor

`- initWithKey:ascending:` (page 8)
Returns an `NSSortDescriptor` object initialized with a given property key path and sort order, and with the default comparison selector.

`- initWithKey:ascending:selector:` (page 8)
Returns an `NSSortDescriptor` object initialized with a given property key path, sort order, and comparison selector.

## Getting Information About a Sort Descriptor

`- ascending` (page 7)
Returns a Boolean value that indicates whether the receiver specifies sorting in ascending order.

`- key` (page 9)
Returns the receiver's property key path.

`- selector` (page 10)
Returns the selector the receiver specifies to use when comparing objects.

## Using Sort Descriptors

`- compareObject:toObject:` (page 7)
Returns an `NSComparisonResult` value that indicates the ordering of two given objects.


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

- `reversedSortDescriptor` (page 9)
  Returns a copy of the receiver with the sort order reversed.

# Instance Methods

## ascending

Returns a Boolean value that indicates whether the receiver specifies sorting in ascending order.

```
- (BOOL)ascending
```

**Return Value**
`YES` if the receiver specifies sorting in ascending order, otherwise `NO`.

**Availability**
Available in Mac OS X v10.3 and later.

**Declared In**
`NSSortDescriptor.h`

## compareObject:toObject:

Returns an `NSComparisonResult` value that indicates the ordering of two given objects.

```
- (NSComparisonResult)compareObject:(id)object1 toObject:(id)object2
```

**Parameters**
*`object1`*
  The object to compare with *`object2`*. This object must have a property accessible using the key-path specified by key (page 9).

  This value must not be `nil`. If the value is `nil`, the behavior is undefined and may change in future versions of Mac OS X.

*`object2`*
  The object to compare with *`object1`*. This object must have a property accessible using the key-path specified by key (page 9).

  This value must not be `nil`. If the value is `nil`, the behavior is undefined and may change in future versions of Mac OS X.

**Return Value**
`NSOrderedAscending` if *`object1`* is less than *`object2`*, `NSOrderedDescending` if *`object1`* is greater than *`object2`*, or `NSOrderedSame` if *`object1`* is equal to *`object2`*.

**Discussion**
The ordering is determined by comparing, using the selector specified selector (page 10), the values of the properties specified by key (page 9) of *`object1`* and *`object2`*.

**Availability**
Available in Mac OS X v10.3 and later.


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

**Declared In**
NSSortDescriptor.h

## initWithKey:ascending:

Returns an NSSortDescriptor object initialized with a given property key path and sort order, and with the default comparison selector.

```
- (id)initWithKey:(NSString *)keyPath ascending:(BOOL)ascending
```

**Parameters**

*keyPath*

The property key to use when performing a comparison. In the comparison, the property is accessed using key-value coding (see *Key-Value Coding Programming Guide*).

*ascending*

YES if the receiver specifies sorting in ascending order, otherwise NO.

**Return Value**
An NSSortDescriptor object initialized with the property key path specified by *keyPath*, sort order specified by *ascending*, and the default comparison selector (compare:).

**Availability**
Available in Mac OS X v10.3 and later.

**See Also**
- initWithKey:ascending:selector: (page 8)

**Related Sample Code**
CoreRecipes
Departments and Employees
NSOperationSample
SimpleCalendar
SpotlightFortunes

**Declared In**
NSSortDescriptor.h

## initWithKey:ascending:selector:

Returns an NSSortDescriptor object initialized with a given property key path, sort order, and comparison selector.

```
- (id)initWithKey:(NSString *)keyPath ascending:(BOOL)ascending
    selector:(SEL)selector
```

**Parameters**

*keyPath*

The property key to use when performing a comparison. In the comparison, the property is accessed using key-value coding (see *Key-Value Coding Programming Guide*).

*ascending*

YES if the receiver specifies sorting in ascending order, otherwise NO.


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

*selector*

The method to use when comparing the properties of objects, for example `caseInsensitiveCompare:` or `localizedCompare:`. The selector must specify a method implemented by the value of the property identified by *keyPath*. The selector used for the comparison is passed a single parameter, the object to compare against `self`, and must return the appropriate `NSComparisonResult` constant. The selector must have the same method signature as:

```
- (NSComparisonResult)localizedCompare:(NSString *)aString
```

**Return Value**
An `NSSortDescriptor` object initialized with the property key path specified by *keyPath*, sort order specified by *ascending*, and the selector specified by *selector*.

**Availability**
Available in Mac OS X v10.3 and later.

**See Also**
`- initWithKey:ascending:` (page 8)

**Related Sample Code**
GridCalendar

**Declared In**
`NSSortDescriptor.h`

## key

Returns the receiver's property key path.

```
- (NSString *)key
```

**Return Value**
The receiver's property key path.

**Discussion**
This key path specifies the property that is compared during sorting.

**Availability**
Available in Mac OS X v10.3 and later.

**Related Sample Code**
iSpend

**Declared In**
`NSSortDescriptor.h`

## reversedSortDescriptor

Returns a copy of the receiver with the sort order reversed.

```
- (id)reversedSortDescriptor
```

**Return Value**
A copy of the receiver with the sort order reversed


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

**Availability**
Available in Mac OS X v10.3 and later.

**Declared In**
`NSSortDescriptor.h`

## selector

Returns the selector the receiver specifies to use when comparing objects.

```
- (SEL)selector
```

**Return Value**
The selector the receiver specifies to use when comparing objects.

**Availability**
Available in Mac OS X v10.3 and later.

**Declared In**
`NSSortDescriptor.h`


2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

# Document Revision History

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

| Date | Notes |
| --- | --- |
| 2007-10-31 | Updated the description of the compareObject:toObject: method. |
| 2006-05-23 | Clarified the comparison mechanism and use of property key, and the description of the selector specified by initWithKey:ascending:selector:. |
| | First publication of this content as a separate document. |



2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History



2007-10-31 | © 2007 Apple Inc. All Rights Reserved.

# Index


## A

`ascending` instance method 7

## C

`compareObject:toObject:` instance method 7

## I

`initWithKey:ascending:` instance method 8
`initWithKey:ascending:selector:` instance method 8

## K

`key` instance method 9

## R

`reversedSortDescriptor` instance method 9

## S

`selector` instance method 10




2007-10-31 | © 2007 Apple Inc. All Rights Reserved.