
NSDateComponents Class Reference

[Cocoa](#) > [Data Management](#)



2008-10-15



Apple Inc.
© 2008 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.

Times is a registered trademark of Heidelberger Druckmaschinen AG, available from Linotype Library GmbH.

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

NSDateComponents Class Reference 5

Overview	5
Tasks	6
Getting Information About an NSDateComponents Object	6
Setting Information for an NSDateComponents Object	6
Instance Methods	7
day	7
era	7
hour	8
minute	8
month	9
second	9
setDay:	9
setEra:	10
setHour:	10
setMinute:	11
setMonth:	11
setSecond:	12
setWeek:	12
setWeekday:	13
setWeekdayOrdinal:	13
setYear:	14
week	14
weekday	15
weekdayOrdinal	15
year	16
Constants	16
NSDateComponents undefined component identifier	16

Document Revision History 17

Index 19

NSDateComponents Class Reference

Inherits from	NSObject
Conforms to	NSCoding NSCopying NSObject (NSObject)
Framework	/System/Library/Frameworks/Foundation.framework
Availability	Available in Mac OS X v10.4 and later.
Companion guide	Date and Time Programming Guide for Cocoa
Declared in	NSCalendar.h
Related sample code	Birthdays Reminders

Overview

`NSDateComponents` encapsulates the components of a date in an extendable, object-oriented manner. It is used to specify a date by providing the temporal components that make up a date and time: hour, minutes, seconds, day, month, year, and so on. It can also be used to specify a duration of time, for example, 5 hours and 16 minutes. An `NSDateComponents` object is not required to define all the component fields. When a new instance of `NSDateComponents` is created the date components are set to `NSUndefinedDateComponent`.

Important: An `NSDateComponents` object is meaningless in itself; you need to know what calendar it is interpreted against, and you need to know whether the values are absolute values of the units, or quantities of the units.

An instance of `NSDateComponents` is not responsible for answering questions about a date beyond the information with which it was initialized. For example, if you initialize one with May 6, 2004, its weekday is `NSUndefinedDateComponent`, not `Thursday`. To get the correct day of the week, you must create a suitable instance of `NSCalendar`, create an `NSDate` object using `dateFromComponents:` and then use `components:fromDate:` to retrieve the weekday—as illustrated in the following example.

```
NSDateComponents *comps = [[NSDateComponents alloc] init];
[comps setDay:6];
[comps setMonth:5];
[comps setYear:2004];
NSCalendar *gregorian = [[NSCalendar alloc]
    initWithCalendarIdentifier:NSGregorianCalendar];
NSDate *date = [gregorian dateFromComponents:comps];
```

```
[comps release];
NSDateComponents *weekdayComponents =
    [gregorian components:NSWeekdayCalendarUnit fromDate:date];
int weekday = [weekdayComponents weekday];
```

For more details, see *Calendars in Date and Time Programming Guide for Cocoa*.

Tasks

Getting Information About an NSDateComponents Object

- [era](#) (page 7)
Returns the number of era units for the receiver.
- [year](#) (page 16)
Returns the number of year units for the receiver.
- [month](#) (page 9)
Returns the number of month units for the receiver.
- [day](#) (page 7)
Returns the number of day units for the receiver.
- [hour](#) (page 8)
Returns the number of hour units for the receiver.
- [minute](#) (page 8)
Returns the number of minute units for the receiver.
- [second](#) (page 9)
Returns the number of second units for the receiver.
- [week](#) (page 14)
Returns the number of week units for the receiver.
- [weekday](#) (page 15)
Returns the number of weekday units for the receiver.
- [weekdayOrdinal](#) (page 15)
Returns the ordinal number of weekday units for the receiver.

Setting Information for an NSDateComponents Object

- [setEra:](#) (page 10)
Sets the number of era units for the receiver.
- [setYear:](#) (page 14)
Sets the number of year units for the receiver.
- [setMonth:](#) (page 11)
Sets the number of month units for the receiver.
- [setDay:](#) (page 9)
Sets the number of day units for the receiver.

- [setHour:](#) (page 10)
Sets the number of hour units for the receiver.
- [setMinute:](#) (page 11)
Sets the number of minute units for the receiver.
- [setSecond:](#) (page 12)
Sets the number of second units for the receiver.
- [setWeek:](#) (page 12)
Sets the number of week units for the receiver.
- [setWeekday:](#) (page 13)
Sets the number of weekday units for the receiver.
- [setWeekdayOrdinal:](#) (page 13)
Sets the ordinal number of weekday units for the receiver.

Instance Methods

day

Returns the number of day units for the receiver.

- (NSInteger)day

Return Value

The number of day units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see *Calendars* in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setDay:](#) (page 9)

Related Sample Code

Birthdays

Declared In

NSCalendar.h

era

Returns the number of era units for the receiver.

- (NSInteger)era

Return Value

The number of era units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setEra:](#) (page 10)

Declared In

NSDateCalendar.h

hour

Returns the number of hour units for the receiver.

- (NSInteger)hour

Return Value

The number of hour units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setHour:](#) (page 10)

Declared In

NSDateCalendar.h

minute

Returns the number of minute units for the receiver.

- (NSInteger)minute

Return Value

The number of minute units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setMinute:](#) (page 11)

Declared In

NSCalendar.h

month

Returns the number of month units for the receiver.

- (NSInteger)month

Return Value

The number of month units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setMonth](#): (page 11)

Declared In

NSCalendar.h

second

Returns the number of second units for the receiver.

- (NSInteger)second

Return Value

The number of second units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setSecond](#): (page 12)

Declared In

NSCalendar.h

setDay:

Sets the number of day units for the receiver.

- (void)setDay:(NSInteger)v

Parameters

v

The number of day units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [day](#) (page 7)

Related Sample Code

Reminders

Declared In

NSDateCalendar.h

setEra:

Sets the number of era units for the receiver.

```
- (void)setEra:(NSInteger)v
```

Parameters

v

The number of era units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see Calendars in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [era](#) (page 7)

Declared In

NSDateCalendar.h

setHour:

Sets the number of hour units for the receiver.

```
- (void)setHour:(NSInteger)v
```

Parameters

v

The number of hour units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [hour](#) (page 8)

Related Sample Code

Reminders

Declared In

NSCalendar.h

setMinute:

Sets the number of minute units for the receiver.

```
- (void)setMinute:(NSInteger)v
```

Parameters

v

The number of minute units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [minute](#) (page 8)

Related Sample Code

Reminders

Declared In

NSCalendar.h

setMonth:

Sets the number of month units for the receiver.

```
- (void)setMonth:(NSInteger)v
```

Parameters

v

The number of month units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [month](#) (page 9)

Related Sample Code

Reminders

Declared In

NSCalendar.h

setSecond:

Sets the number of second units for the receiver.

```
- (void)setSecond:(NSInteger)v
```

Parameters

v

The number of second units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [second](#) (page 9)

Related Sample Code

Reminders

Declared In

NSCalendar.h

setWeek:

Sets the number of week units for the receiver.

```
- (void)setWeek:(NSInteger)v
```

Parameters

v

The number of week units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [week](#) (page 14)

Declared In

NSCalendar.h

setWeekday:

Sets the number of weekday units for the receiver.

```
- (void)setWeekday:(NSInteger)v
```

Parameters

v

The number of weekday units for the receiver.

Discussion

Weekday units are the numbers 1 through *n*, where *n* is the number of days in the week. For example, in the Gregorian calendar, *n* is 7 and Sunday is represented by 1.

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

– [weekday](#) (page 15)

Declared In

NSCalendar.h

setWeekdayOrdinal:

Sets the ordinal number of weekday units for the receiver.

```
- (void)setWeekdayOrdinal:(NSInteger)v
```

Parameters

v

The ordinal number of weekday units for the receiver.

Discussion

Weekday ordinal units represent the position of the weekday within the next larger calendar unit, such as the month. For example, 2 is the weekday ordinal unit for the *second* Friday of the month.

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [weekdayOrdinal](#) (page 15)

Declared In

NSCalendar.h

setYear:

Sets the number of year units for the receiver.

```
- (void)setYear:(NSInteger)v
```

Parameters

v

The number of year units for the receiver.

Discussion

This value will be interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [year](#) (page 16)

Related Sample Code

Reminders

Declared In

NSCalendar.h

week

Returns the number of week units for the receiver.

```
- (NSInteger)week
```

Return Value

The number of week units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setWeek:](#) (page 12)

Declared In

NSDateCalendar.h

weekday

Returns the number of weekday units for the receiver.

- (NSInteger)weekday

Return Value

The number of weekday units for the receiver.

Discussion

Weekday units are the numbers 1 through n , where n is the number of days in the week. For example, in the Gregorian calendar, n is 7 and Sunday is represented by 1.

This value is interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setWeekday:](#) (page 13)

Related Sample Code

Birthdays

Declared In

NSDateCalendar.h

weekdayOrdinal

Returns the ordinal number of weekday units for the receiver.

- (NSInteger)weekdayOrdinal

Return Value

The ordinal number of weekday units for the receiver.

Discussion

Weekday ordinal units represent the position of the weekday within the next larger calendar unit, such as the month. For example, 2 is the weekday ordinal unit for the *second* Friday of the month.

This value is interpreted in the context of the calendar with which it is used—see *Calendars in Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setWeekdayOrdinal:](#) (page 13)

Declared In

NSCalendar.h

year

Returns the number of year units for the receiver.

- (NSInteger)year

Return Value

The number of year units for the receiver.

Discussion

This value is interpreted in the context of the calendar with which it is used—see *Calendars* in *Date and Time Programming Guide for Cocoa*.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [setYear:](#) (page 14)

Declared In

NSCalendar.h

Constants

NSDateComponents undefined component identifier

This constant specifies that an `NSDateComponents` component is undefined.

```
enum {
    NSUndefinedDateComponent = 0x7fffffff
};
```

Constants

NSUndefinedDateComponent

Specifies that the component is undefined.

Available in Mac OS X v10.4 and later.

Declared in `NSCalendar.h`.

Declared In

NSCalendar.h

Document Revision History

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

Date	Notes
2008-10-15	Clarified meaning of weekday and weekday ordinal units.
2007-10-31	Added links from discussions to the relevant conceptual material.
2006-05-23	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

D

day [instance method 7](#)

E

era [instance method 7](#)

H

hour [instance method 8](#)

M

minute [instance method 8](#)

month [instance method 9](#)

N

NSDateComponents undefined component identifier [16](#)

NSUndefinedDateComponent [constant 16](#)

S

second [instance method 9](#)

setDay: [instance method 9](#)

setEra: [instance method 10](#)

setHour: [instance method 10](#)

setMinute: [instance method 11](#)

setMonth: [instance method 11](#)

setSecond: [instance method 12](#)

setWeek: [instance method 12](#)

setWeekday: [instance method 13](#)

setWeekdayOrdinal: [instance method 13](#)

setYear: [instance method 14](#)

W

week [instance method 14](#)

weekday [instance method 15](#)

weekdayOrdinal [instance method 15](#)

Y

year [instance method 16](#)