
ABIdentityPicker Class Objective-C Reference

[Cocoa](#) > [Apple Applications](#)



2006-08-23



Apple Inc.
© 2006 Apple Computer, 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, and Objective-C are trademarks of Apple Inc., registered in the United States and other countries.

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

ABIdentityPicker Class Objective-C Reference 5

Overview	5
Tasks	5
Initializing	5
Setting and Getting Properties	5
Running an Identity Picker	6
Instance Methods	6
allowsMultipleSelection	6
beginSheetModalForWindow:modalDelegate:didEndSelector:contextInfo:	6
init	7
runModalIdentityPicker	7
setAllowsMultipleSelection:	7
Constants	8
Identity Constants	8

Document Revision History 9

Index 11

ABIdentityPicker Class Objective-C Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/AddressBook.framework
Declared in	ABIdentityPicker.h
Companion guides	Address Book Programming Guide for Mac OS X Identity Services Programming Guide

Overview

An `ABIdentityPicker` object allows a user to select Address Book records—for example, group or person objects—that it wants one or more services or shared resources to have access to. An identity picker can be displayed either as an application-modal dialog or as a sheet attached to a document window. An identity picker returns the selected records to be added to access control lists using Identity Services. If a selected record is not a sharing account, then an identity picker prompts the end user for additional information—such as a password—to promote that record to a sharing account.

Tasks

Initializing

- [init](#) (page 7)
Initializes the receiver immediately after memory for it has been allocated.

Setting and Getting Properties

- [setAllowsMultipleSelection:](#) (page 7)
Allows a user to make multiple selections.
- [allowsMultipleSelection](#) (page 6)
Returns a Boolean value indicating whether the user is allowed to select multiple records.

Running an Identity Picker

- [runModalIdentityPicker](#) (page 7)
Runs the receiver as an application-modal dialog and returns the selected records.
- [beginSheetModalForWindow:modalDelegate:didEndSelector:contextInfo:](#) (page 6)
Runs the receiver modally as a sheet attached to a specified window.

Instance Methods

allowsMultipleSelection

Returns a Boolean value indicating whether the user is allowed to select multiple records.

- (BOOL)allowsMultipleSelection

Return Value

YES if the user can select multiple records; otherwise, NO.

See Also

- [setAllowsMultipleSelection:](#) (page 7)

beginSheetModalForWindow:modalDelegate:didEndSelector:contextInfo:

Runs the receiver modally as a sheet attached to a specified window.

- (void)beginSheetModalForWindow:(NSWindow *)*window* modalDelegate:(id)*delegate* didEndSelector:(SEL)*didEndSelector* contextInfo:(void *)*contextInfo*

Parameters

window

The parent window for the sheet.

delegate

The delegate for the modal session.

didEndSelector

A message sent to the delegate after the user responds but before the sheet is dismissed.

contextInfo

Contextual data passed to the delegate in the *didEndSelector* message.

Discussion

The *didEndSelector* argument must be a selector that takes three arguments, and the corresponding method should have a declaration modeled on the following example:

- (void)identityPickerDidEnd:(ABIdentityPicker *)identityPickerController identities:(NSArray *)identities contextInfo:(void *)contextInfo;

where the *identityPickerController* argument is the identity picker object, the *identities* argument is an array containing ABPerson objects, and *contextInfo* is the same *contextInfo* that was passed in the original message.

See Also

- [runModalIdentityPicker](#) (page 7)
- `identityUniqueId`
- `identity`

init

Initializes the receiver immediately after memory for it has been allocated.

- `(id)init`

runModalIdentityPicker

Runs the receiver as an application-modal dialog and returns the selected records.

- `(NSArray *)runModalIdentityPicker`

Return Value

An array containing the ABRecord objects that the user selected in the panel.

Discussion

The receiver may create identities for selected records if necessary.

See Also

- [beginTransactionForWindow:modalDelegate:didEndSelector:contextInfo:](#) (page 6)
- `identityUniqueId`
- `identity`

setAllowsMultipleSelection:

Allows a user to make multiple selections.

- `(void)setAllowsMultipleSelection:(BOOL)flag`

Parameters

flag

YES if you can select multiple records; otherwise, NO.

Discussion

By default, you cannot select multiple records.

See Also

- [allowsMultipleSelection](#) (page 6)

Constants

Identity Constants

Constants used to identify objects.

```
extern NSString * const kABIdentityUIDProperty;
```

Constants

`kABIdentityUIDProperty`

The identity UID property.

Declared In

`ABIdentityPicker.h`

Document Revision History

This table describes the changes to *ABIdentityPicker Class Objective-C Reference*.

Date	Notes
2006-08-23	New document that describes the identity picker, a dialog or sheet used to select groups and people for Identity Services.

REVISION HISTORY

Document Revision History

Index

A

`allowsMultipleSelection` **instance method** [6](#)

B

`beginSheetModalForWindow:modalDelegate:
didEndSelector:contextInfo:` **instance method**
[6](#)

I

Identity Constants [8](#)
`init` **instance method** [7](#)

K

`kABIdentityUIDProperty` **constant** [8](#)

R

`runModalIdentityPicker` **instance method** [7](#)

S

`setAllowsMultipleSelection:` **instance method** [7](#)