
WebScriptObject Class Reference

[Cocoa](#) > [User Experience](#)



2009-04-08



Apple Inc.
© 2009 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, Mac OS, and Objective-C are trademarks of Apple Inc., registered in the United States and other countries.

WebScript is a trademark of Apple Inc.

Java and all Java-based trademarks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. 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

WebScriptObject Class Reference 5

Overview	5
Tasks	6
Getting and Setting Properties	6
Executing Scripts	6
Raising Exceptions	6
Getting a String Representation	6
Class Methods	6
throwException:	6
Instance Methods	7
callWebScriptMethod:withArguments:	7
evaluateWebScript:	7
JSObject	8
removeWebScriptKey:	8
setException:	9
setWebScriptValueAtIndex:value:	9
stringRepresentation	9
webScriptValueAtIndex:	10

Document Revision History 11

Index 13

WebScriptObject Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/WebKit.framework
Availability	Available in Mac OS X v10.3.9 and later.
Companion guide	WebKit Objective-C Programming Guide
Declared in	WebScriptObject.h
Related sample code	Birthdays CallJS Fortune Reminders

Overview

A `WebScriptObject` object is an Objective-C wrapper for a scripting object passed to your application from the scripting environment.

You can not create a `WebScriptObject` object directly. You get a window `WebScriptObject` object by sending `windowScriptObject` to your `WebView` object.

You can use key-value coding methods—for example, `setValue:forKey:` and `valueForKey:`—to get and set properties of a `WebScriptObject` object. You can also access properties by index using the [setWebScriptValueAtIndex:value:](#) (page 9) and [webScriptValueAtIndex:](#) (page 10) methods. Use the [removeWebScriptKey:](#) (page 8) method to remove a scripting object property.

Not all properties and methods of a class are exported. Use the `setValue:forUndefinedKey:` and `valueForUndefinedKey:` methods to intercept access to properties that are not exported. Similarly, use the `invokeUndefinedMethodFromWebScript:withArguments:` method to intercept method invocations that are not exported.

If you want access to properties and methods defined in your own classes, use the methods in the `WebScripting` informal protocol to specify the properties and methods the class should export to WebKit's JavaScript environment.

Use the [callWebScriptMethod:withArguments:](#) (page 7) and [evaluateWebScript:](#) (page 7) methods to execute scripts in the scripting environment.

Tasks

Getting and Setting Properties

- [JSObject](#) (page 8)
Returns the JavaScript object corresponding to the receiver.
- [removeWebScriptKey:](#) (page 8)
Removes a property from a scripting environment.
- [webScriptValueAtIndex:](#) (page 10)
Returns the value of a property at the specified index.
- [setWebScriptValueAtIndex:value:](#) (page 9)
Sets the value of a property at the specified index.

Executing Scripts

- [callWebScriptMethod:withArguments:](#) (page 7)
Returns the result of executing a method in the scripting environment.
- [evaluateWebScript:](#) (page 7)
Returns the result of evaluating a script in the scripting environment.

Raising Exceptions

- + [throwException:](#) (page 6)
Raises an exception in the current script execution context.
- [setException:](#) (page 9)
Raises a scripting environment exception in the context of the current object.

Getting a String Representation

- [stringRepresentation](#) (page 9)
Returns a string representation of the receiver.

Class Methods

throwException:

Raises an exception in the current script execution context.

+ (BOOL)throwException:(NSString *)*exceptionMessage*

Parameters

exceptionMessage

The exception message.

Return Value

YES if successful, NO otherwise.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

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

Declared In

WebScriptObject.h

Instance Methods

callWebScriptMethod:withArguments:

Returns the result of executing a method in the scripting environment.

```
- (id)callWebScriptMethod:(NSString *)namewithArguments:(NSArray *)args
```

Parameters

name

The name of the method to invoke.

args

The values to pass to the method.

Return Value

The return value of the method. Returns `WebUndefined` if an exception is thrown in the JavaScript environment or the method has no return value.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

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

Related Sample Code

QT Capture Widget

WebKitPluginWithJavaScript

Declared In

WebScriptObject.h

evaluateWebScript:

Returns the result of evaluating a script in the scripting environment.

```
- (id)evaluateWebScript:(NSString *)script
```

Parameters

script

The script to evaluate.

Return Value

The scripting object. The format of the script is dependent on the target scripting environment. Returns `WebUndefined` if an exception is thrown in the JavaScript environment or there is no return value.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

- [callWebScriptMethod:withArguments:](#) (page 7)

Related Sample Code

Reminders

Declared In

WebScriptObject.h

JSObject

Returns the JavaScript object corresponding to the receiver.

```
- (JSObjectRef)JSObject
```

Return Value

The JavaScript object corresponding to the receiver in the JavaScriptCore C API.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebScriptObject.h

removeWebScriptKey:

Removes a property from a scripting environment.

```
- (void)removeWebScriptKey:(NSString *)name
```

Parameters

name

Property to remove.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

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

- [setWebScriptValueAtIndex:value:](#) (page 9)

Declared In

WebScriptObject.h

setException:

Raises a scripting environment exception in the context of the current object.

- (void)setException:(NSString *)*description*

Parameters

description

Description of the exception.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

+ [throwException:](#) (page 6)

Declared In

WebScriptObject.h

setWebScriptValueAtIndex:value:

Sets the value of a property at the specified index.

- (void)setWebScriptValueAtIndex:(unsigned)*index*value:(id)*value*

Parameters

index

The index of the property.

value

The value of the property.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

- [removeWebScriptKey:](#) (page 8)

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

Declared In

WebScriptObject.h

stringRepresentation

Returns a string representation of the receiver.

- (NSString *)stringRepresentation

Return Value

The string representation of the receiver.

Discussion

The coercion of nonstring objects is dependent on the scripting environment.

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebScriptObject.h

webScriptValueAtIndex:

Returns the value of a property at the specified index.

- (id)webScriptValueAtIndex:(unsigned) *index*

Parameters

index

The index of the property.

Return Value

The value of a property at *index*. Returns `WebUndefined` if an exception is thrown in the JavaScript environment.

Discussion

Accessing property values by index is dependent on the scripting environment.

Availability

Available in Mac OS X v10.3.9 and later.

See Also

- [removeWebScriptKey:](#) (page 8)
- [setWebScriptValueAtIndex:value:](#) (page 9)

Declared In

WebScriptObject.h

Document Revision History

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

Date	Notes
2009-04-08	Added separate parameter and return value descriptions.
2008-10-15	Minor edits throughout.
2008-06-19	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

C

callWebScriptMethod:withArguments: **instance method** [7](#)

E

evaluateWebScript: **instance method** [7](#)

J

JSObject **instance method** [8](#)

R

removeWebScriptKey: **instance method** [8](#)

S

setException: **instance method** [9](#)
setWebScriptValueAtIndex:value: **instance method** [9](#)
stringRepresentation **instance method** [9](#)

T

throwException: **class method** [6](#)

W

webScriptValueAtIndex: **instance method** [10](#)