
WebUIDelegate Protocol Reference

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



2009-03-04



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, Objective-C, Pages, and Safari are trademarks of Apple Inc., registered in the United States and other countries.

Spotlight 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

WebUIDelegate Protocol Reference 5

Overview	5
Tasks	5
Creating and Closing Windows	5
Moving and Resizing Windows	5
Moving and Resizing Content Views	6
Making Windows Key and Main	6
Ordering Windows	6
Working with the Responder Chain	6
Handling Mouse Events	6
Opening Panels	7
Displaying Status Messages	7
Managing Toolbars and the Status Bar	7
Controlling Drag Behavior	8
Controlling Other Behaviors	8
Printing	8
Instance Methods	8
webView:contextMenuItemsForElement:defaultMenuItems:	8
webView:createWebViewModalDialogWithRequest:	9
webView:createWebViewWithRequest:	10
webView:dragDestinationActionMaskForDraggingInfo:	10
webView:dragSourceActionMaskForPoint:	11
webView:drawFooterInRect:	11
webView:drawHeaderInRect:	12
webView:makeFirstResponder:	12
webView:mouseDidMoveOverElement:modifierFlags:	13
webView:printFrameView:	13
webView:runBeforeUnloadConfirmPanelWithMessage:initiatedByFrame:	14
webView:runJavaScriptAlertPanelWithMessage:initiatedByFrame:	14
webView:runJavaScriptConfirmPanelWithMessage:initiatedByFrame:	15
webView:runJavaScriptTextInputPanelWithPrompt:defaultText:initiatedByFrame:	15
webView:runOpenPanelForFileButtonWithResultListener:	16
webView:setFrame:	16
webView:setResizable:	17
webView:setStatusBarVisible:	17
webView:setStatusText:	18
webView:setToolbarsVisible:	18
webView:shouldPerformAction:fromSender:	19
webView:validateUserInterfaceItem:defaultValidation:	19
webView:willPerformDragDestinationAction:forDraggingInfo:	20
webView:willPerformDragSourceAction:fromPoint:withPasteboard:	21

- webViewAreToolbarsVisible: 21
- webViewClose: 22
- webViewFirstResponder: 22
- webViewFocus: 22
- webViewFooterHeight: 23
- webViewFrame: 23
- webViewHeaderHeight: 24
- webViewIsResizable: 24
- webViewIsStatusBarVisible: 25
- webViewRunModal: 25
- webViewShow: 26
- webViewStatusText: 26
- webViewUnfocus: 27
- Constants 27
 - Menu Item Tags 27
 - Drag-Destination Actions 31
 - Drag-Source Actions 32

Appendix A Deprecated WebUIDelegate Methods 35

- Deprecated in Mac OS X v10.4.11 35
 - webView:runJavaScriptAlertPanelWithMessage: 35
 - webView:runJavaScriptConfirmPanelWithMessage: 35
 - webView:runJavaScriptTextInputPanelWithPrompt:defaultText: 36
 - webView:setContentRect: 37
 - webViewContentRect: 37

Document Revision History 39

Index 41

WebUIDelegate Protocol Reference

(informal protocol)

Framework	/System/Library/Frameworks/WebKit.framework
Availability	Available in Mac OS X v10.2 with Safari 1.0 and later. Available in Mac OS X v10.2.7 and later.
Companion guide	WebKit Objective-C Programming Guide
Declared in	WebUIDelegate.h

Overview

Web view user interface delegates implement this informal protocol to control the opening of new windows, augment the behavior of default menu items displayed when the user clicks elements, and perform other user interface–related tasks. These methods can be invoked as a result of handling JavaScript or other plug-in content. Delegates that display more than one web view per window, for example, need to implement some of these methods to handle that case. The default implementation assumes one window per web view, so non-conventional user interfaces might implement a user interface delegate.

Tasks

Creating and Closing Windows

- [webView:createWebViewModalDialogWithRequest:](#) (page 9)
Creates a modal window containing a web view that loads the specified request.
- [webViewRunModal:](#) (page 25)
Displays a web view in a modal window.
- [webView:createWebViewWithRequest:](#) (page 10)
Creates a window containing a web view to load the specified request.
- [webViewClose:](#) (page 22)
Closes a web view in a window.

Moving and Resizing Windows

- [webViewIsResizable:](#) (page 24)
Returns a Boolean value indicating whether a web view’s window can be resized.

- [webView:setResizable:](#) (page 17)
Sets whether a web view's window can be resized.
- [webView:setFrame:](#) (page 16)
Sets the frame rectangle of a web view's window to the specified frame size.
- [webViewFrame:](#) (page 23)
Returns the frame rectangle of a web view's window.

Moving and Resizing Content Views

- [webView:setContentRect:](#) (page 37) **Deprecated in Mac OS X v10.4.11**
Sets the window's content view frame to the specified content rectangle. (**Deprecated.** Content rectangle calculations are automatic.)
- [webViewContentRect:](#) (page 37) **Deprecated in Mac OS X v10.4.11**
Returns a web view window's content rectangle. (**Deprecated.** Content rectangle calculations are automatic.)

Making Windows Key and Main

- [webViewFocus:](#) (page 22)
Brings a web view's window to the front and makes it the active window.
- [webViewUnfocus:](#) (page 27)
Relinquishes focus on a web view's window.

Ordering Windows

- [webViewShow:](#) (page 26)
Displays a web view's window and moves it to the front.

Working with the Responder Chain

- [webViewFirstResponder:](#) (page 22)
Returns the first responder of the web view's window.
- [webView:makeFirstResponder:](#) (page 12)
Sets the first responder of a web view's window to the specified view.

Handling Mouse Events

- [webView:mouseDidMoveOverElement:modifierFlags:](#) (page 13)
Updates information about the element the user is mousing over.
- [webView:contextMenuItemsForElement:defaultMenuItems:](#) (page 8)
Returns menu items to display in an element's contextual menu.

Opening Panels

- [webView:runJavaScriptAlertPanelWithMessage:initiatedByFrame:](#) (page 14)
Displays a JavaScript alert panel containing the specified message.
- [webView:runJavaScriptConfirmPanelWithMessage:initiatedByFrame:](#) (page 15)
Displays a JavaScript confirmation panel with the specified message.
- [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:initiatedByFrame:](#) (page 15)
Displays a JavaScript text input panel and returns the entered text.
- [webView:runOpenPanelForFileButtonWithResultListener:](#) (page 16)
Displays an open panel for a file input control.
- [webView:runBeforeUnloadConfirmPanelWithMessage:initiatedByFrame:](#) (page 14)
Displays a confirmation panel containing the specified message before a window closes.
- [webView:runJavaScriptAlertPanelWithMessage:](#) (page 35) **Deprecated in Mac OS X v10.4.11**
Displays a JavaScript alert panel. (**Deprecated.** Use [webView:runJavaScriptAlertPanelWithMessage:initiatedByFrame:](#) (page 14) instead.)
- [webView:runJavaScriptConfirmPanelWithMessage:](#) (page 35) **Deprecated in Mac OS X v10.4.11**
Displays a JavaScript confirm panel. (**Deprecated.** Use [webView:runJavaScriptConfirmPanelWithMessage:initiatedByFrame:](#) (page 15) instead.)
- [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:](#) (page 36) **Deprecated in Mac OS X v10.4.11**
Displays a JavaScript text input panel and returns the entered text. (**Deprecated.** Use [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:initiatedByFrame:](#) (page 15) instead.)

Displaying Status Messages

- [webView:setStatusText:](#) (page 18)
Sets the status message displayed by a web view's window, if any, to the specified text.
- [webViewStatusText:](#) (page 26)
Returns the current status message from a web view's window.

Managing Toolbars and the Status Bar

- [webViewAreToolbarsVisible:](#) (page 21)
Returns a Boolean value indicating whether any toolbars are visible in a web view's window.
- [webView:setToolbarsVisible:](#) (page 18)
Sets whether a web view's toolbars should be visible.
- [webViewIsStatusBarVisible:](#) (page 25)
Returns a Boolean value indicating whether the status bar in a web view's window is visible.
- [webView:setStatusBarVisible:](#) (page 17)
Sets the visibility of the status bar in a web view's window.

Controlling Drag Behavior

- [webView:dragDestinationActionMaskForDraggingInfo:](#) (page 10)
Returns a mask indicating which drag operations are allowed by the sender.
- [webView:dragSourceActionMaskForPoint:](#) (page 11)
Returns a mask indicating which drag-source actions are allowed for a drag that begins at the specified location.
- [webView:willPerformDragDestinationAction:forDraggingInfo:](#) (page 20)
Tells the receiver that the sending web view will perform the specified drag-destination action.
- [webView:willPerformDragSourceAction:fromPoint:withPasteboard:](#) (page 21)
Tells the receiver that the sending web view will perform the specified drag-source action.

Controlling Other Behaviors

- [webView:shouldPerformAction:fromSender:](#) (page 19)
Returns a Boolean value that indicates whether the action sent by the specified object should be performed.
- [webView:validateUserInterfaceItem:defaultValidation:](#) (page 19)
Returns a Boolean value that indicates whether the specified user interface item is valid.

Printing

- [webView:printFrameView:](#) (page 13)
Prints the contents of a web frame view.
- [webViewHeaderHeight:](#) (page 24)
Returns the height of the web view's printed page header.
- [webViewFooterHeight:](#) (page 23)
Returns the height of the web view's printed page footer.
- [webView:drawHeaderInRect:](#) (page 12)
Draws the web view's header in the specified rectangle.
- [webView:drawFooterInRect:](#) (page 11)
Draws the web view's footer in the specified rectangle.

Instance Methods

webView:contextMenuItemsForElement:defaultMenuItems:

Returns menu items to display in an element's contextual menu.

- (NSArray *)**webView:(WebView *)sender contextMenuItemsForElement:(NSDictionary *)element defaultMenuItems:(NSArray *)defaultMenuItems**

Parameters*sender*

The web view that sent the message.

element

A dictionary that describes the element that was clicked. See “Constants” in *WebView Class Reference* for information about the key-value pairs in this dictionary.

defaultMenuItems

The menu items included by default in the element’s contextual menu. See “Menu Item Tags” (page 27) for values you can use to differentiate among specific menu items.

Return Value

An array of menu items to display in the element’s contextual menu.

Discussion

This method is invoked every time the user clicks the right mouse button, or control-clicks, on an element to reveal a contextual menu. The receiver typically returns a modified copy of the default menu items dictionary, adding and removing menu items as appropriate for this type of element. You can use this mechanism to remove items that are not appropriate for a particular environment or task, such as saving files to the desktop in a web kiosk. You do not need to set the actions and targets of the default items.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

WebUIDelegate.h

webView:createWebViewModalDialogWithRequest:

Creates a modal window containing a web view that loads the specified request.

```
- (WebView *)webView:(WebView *)sender
  createWebViewModalDialogWithRequest:(NSURLRequest *)request
```

Parameters*sender*

The web view that sent the message.

request

The request to load.

Return Value

The web view that is loading the specified request.

Discussion

This method is invoked when JavaScript calls `window.showModalDialog`. It should create a new modal window containing the web view and initially hide the window. The `webViewRunModal:` (page 25) message is sent to the delegate to display the web view.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webView:createWebViewWithRequest:

Creates a window containing a web view to load the specified request.

```
- (WebView *)webView:(WebView *)sender createWebViewWithRequest:(NSURLRequest *)request
```

Parameters

sender

The web view that sent the message.

request

The request to load.

Return Value

The web view that is loading the request.

Discussion

This method should begin loading the content for the specified request by sending `loadRequest:` to its main frame. The new window should initially be hidden. Later, a `webViewShow:` (page 26) message is sent to the delegate of the new web view. By default, this method returns `nil`.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

WebUIDelegate.h

webView:dragDestinationActionMaskForDraggingInfo:

Returns a mask indicating which drag operations are allowed by the sender.

```
- (NSUInteger)webView:(WebView *)sender dragDestinationActionMaskForDraggingInfo:(id <NSDraggingInfo>)draggingInfo
```

Parameters

sender

The web view that sent the message.

draggingInfo

The information object for the dragging operation.

Return Value

A mask that indicates which drag operations are allowed when content is dragged over the sending web view. (Note that the return value changed from an `unsigned int` to an `NSUInteger` in Mac OS X v10.5.) See “[Drag-Destination Actions](#)” (page 31) for a list of return values.

Discussion

This method can be invoked multiple times while content is dragged over the sending web view. When the content is dropped, the web view sends a notification (`webView:willPerformDragDestinationAction:forDraggingInfo:` (page 20)) to the receiver.

If you do not implement this method, it returns `WebDragDestinationActionAny` (page 32) by default.

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webView:dragSourceActionMaskForPoint:

Returns a mask indicating which drag-source actions are allowed for a drag that begins at the specified location.

```
- (NSUInteger)webView:(WebView *)sender dragSourceActionMaskForPoint:(NSPoint)point
```

Parameters*sender*

The web view that sent the message.

point

The point at which the drag began, specified in the coordinates of the web view.

Return Value

A mask indicating which drag-source actions are allowed. (Note that the return value changed from an unsigned int to an NSUInteger in Mac OS X v10.5.) See “[Drag-Source Actions](#)” (page 32) for a list of return values.

Discussion

This method is called after the user has begun a drag from a point in a web view. This method can be invoked multiple times while content is dragged from the sending web view. When the content is dropped, the sender sends [webView:willPerformDragSourceAction:fromPoint:withPasteboard:](#) (page 21) to the receiver.

If you do not implement this method, it returns ([WebDragSourceActionAny](#) & [~WebDragSourceActionLink](#)) if the cursor is in an editable part of the web view; otherwise, it returns [WebDragDestinationActionAny](#) (page 32).

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webView:drawFooterInRect:

Draws the web view’s footer in the specified rectangle.

```
- (void)webView:(WebView *)sender drawFooterInRect:(NSRect)rect
```

Parameters*sender*

The web view that sent the message.

rect

The rectangle reserved for drawing the footer.

Availability

Available in Mac OS X v10.4.11 and later.

See Also

- [webViewFooterHeight:](#) (page 23)

Declared In

WebUIDelegate.h

webView:drawHeaderInRect:

Draws the web view's header in the specified rectangle.

```
- (void)webView:(WebView *)sender drawHeaderInRect:(NSRect)rect
```

Parameters

sender

The web view that sent the message.

rect

The rectangle reserved for drawing the header.

Availability

Available in Mac OS X v10.4.11 and later.

See Also

- [webViewHeaderHeight:](#) (page 24)

Declared In

WebUIDelegate.h

webView:makeFirstResponder:

Sets the first responder of a web view's window to the specified view.

```
- (void)webView:(WebView *)sender makeFirstResponder:(NSResponder *)responder
```

Parameters

sender

The web view that sent the message.

responder

A view in the web view's hierarchy.

Discussion

You can ignore this message if *sender* is not yet attached to a window.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webViewFirstResponder:](#) (page 22)

Declared In

WebUIDelegate.h

webView:mouseDidMoveOverElement:modifierFlags:

Updates information about the element the user is mousing over.

```
- (void)webView:(WebView *)sender mouseDidMoveOverElement:(NSDictionary *)elementInformation modifierFlags:(NSUInteger)modifierFlags
```

Parameters

sender

The web view that sent the message.

elementInformation

A dictionary that describes the element under the mouse, or `nil`. See “Constants” in *WebView Class Reference* for information about the key-value pairs in this dictionary.

modifierFlags

An integer bit field that indicates the modifier keys in effect during the event. See “Modifier Flags” in *NSEvent Class Reference* for information about possible modifiers. Note that this parameter was changed from an unsigned `int` to an `NSUInteger` in Mac OS X v10.5.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

WebUIDelegate.h

webView:printFrameView:

Prints the contents of a web frame view.

```
- (void)webView:(WebView *)sender printFrameView:(WebFrameView *)frameView
```

Parameters

sender

The web view that sent the message.

frameView

The web frame view whose contents to print.

Discussion

This method is invoked when a script or a user wants to print a webpage. Typically, the delegate implements this method to prepare the web frame view content for printing. The web frame view can handle some content without intervention by the delegate. Send the `documentViewShouldHandlePrint` message to the web frame view to determine if it can handle printing. If this method returns `YES`, then the delegate can print the content by sending the `printDocumentView` message to the web frame view. Otherwise, the delegate can use `printOperationWithPrintInfo:` to get an `NSPrintOperation` object to print the web frame view.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webView:runBeforeUnloadConfirmPanelWithMessage:initiatedByFrame:

Displays a confirmation panel containing the specified message before a window closes.

```
- (BOOL)webView:(WebView *)sender runBeforeUnloadConfirmPanelWithMessage:(NSString *)message initiatedByFrame:(WebFrame *)frame
```

Parameters

sender

The web view that sent the message.

message

The message to display in the panel.

frame

The web frame whose JavaScript initiated this call.

Return Value

YES if the user clicked the OK button; otherwise, NO.

Discussion

Use this method to include a message in the confirmation panel in addition to the message supplied by the webpage. The confirmation panel should contain OK and Cancel buttons.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webView:runJavaScriptAlertPanelWithMessage:initiatedByFrame:

Displays a JavaScript alert panel containing the specified message.

```
- (void)webView:(WebView *)sender runJavaScriptAlertPanelWithMessage:(NSString *)message initiatedByFrame:(WebFrame *)frame
```

Parameters

sender

The web view that sent the message.

message

The message to display in the alert panel.

frame

The web frame whose JavaScript initiated this call.

Discussion

This method displays an alert panel when JavaScript code calls `alert`. Delegates should visually indicate that this panel comes from JavaScript. The panel should contain a single OK button. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webView:runJavaScriptConfirmPanelWithMessage:initiatedByFrame:

Displays a JavaScript confirmation panel with the specified message.

```
- (BOOL)webView:(WebView *)sender runJavaScriptConfirmPanelWithMessage:(NSString *)message initiatedByFrame:(WebFrame *)frame
```

Parameters

sender

The web view that sent the message.

message

The message to display in the confirmation panel.

frame

The web frame whose JavaScript initiated this call.

Return Value

YES if the user clicks OK; otherwise, NO.

Discussion

This method displays a confirmation panel when JavaScript code calls `confirm`. Delegates should visually indicate that this panel comes from JavaScript. The panel should contain an OK and a Cancel button. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webView:runJavaScriptTextInputPanelWithPrompt:defaultText:initiatedByFrame:

Displays a JavaScript text input panel and returns the entered text.

```
- (NSString *)webView:(WebView *)sender runJavaScriptTextInputPanelWithPrompt:(NSString *)prompt defaultText:(NSString *)defaultText initiatedByFrame:(WebFrame *)frame
```

Parameters

sender

The web view that sent the message.

prompt

The message to display in the text input panel.

defaultText

Default placeholder text to display in the text field.

frame

The web frame whose JavaScript initiated this call.

Return Value

The text entered by the user if the user clicks OK; otherwise, `nil`.

Discussion

This method is used to provide an alternate text input panel when JavaScript code calls `prompt`. Delegates should visually indicate that this panel comes from JavaScript. The panel should contain an OK and a Cancel button, and an editable text field. If you do not implement this method, a JavaScript text input panel is displayed.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

`WebUIDelegate.h`

webView:runOpenPanelForFileButtonWithResultListener:

Displays an open panel for a file input control.

```
- (void)webView:(WebView *)sender runOpenPanelForFileButtonWithResultListener:(id
    < WebOpenPanelResultListener >)resultListener
```

Parameters

sender

The web view that sent the message.

resultListener

See the `WebOpenPanelResultListener` protocol for how to set these values.

Discussion

This method uses a listener object to set the results of the open panel, instead of returning the value directly. This approach allows delegates to implement the open panel as a modal dialog. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

`WebUIDelegate.h`

webView:setFrame:

Sets the frame rectangle of a web view's window to the specified frame size.

```
- (void)webView:(WebView *)sender setFrame:(NSRect)frame
```

Parameters

sender

The web view that sent the message.

frame

The frame size.

Discussion

The sender invokes this method instead of setting the window's frame directly, allowing delegates to augment the behavior by, for example, saving the original window size before resizing as a result of JavaScript running. If you do not implement this method, the `NSWindow` method `setFrame:display:` is sent to the window that contains *sender*, with `YES` passed as the display argument.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webViewFrame:](#) (page 23)

Declared In

WebUIDelegate.h

webView:setResizable:

Sets whether a web view's window can be resized.

```
- (void)webView:(WebView *)sender setResizable:(BOOL)resizable
```

Parameters

sender

The web view that sent the message.

resizable

If `YES`, the web view's window can be resized; if `NO`, the window is not resizable.

Discussion

By default, this method sets the window containing a web view to be resizable. If you display multiple web views in a window then your user interface delegate should implement this method to handle this special case. If you do not implement this method, the `NSWindow` method `setShowsResizeIndicator:` is sent to the window that contains *sender*.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webViewIsResizable:](#) (page 24)

Declared In

WebUIDelegate.h

webView:setStatusBarVisible:

Sets the visibility of the status bar in a web view's window.

```
- (void)webView:(WebView *)sender setStatusBarVisible:(BOOL)visible
```

Parameters*sender*

The web view that sent the message.

visible

If YES, the delegate should display the status bar (if any); if NO, the delegate should hide the status bar.

Discussion

No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also- [webViewIsStatusBarVisible:](#) (page 25)**Declared In**

WebUIDelegate.h

webView:setStatusText:

Sets the status message displayed by a web view's window, if any, to the specified text.

- (void)webView:(WebView *)sender setStatusText:(NSString *)text

Parameters*sender*

The web view that sent the message.

text

The status message to display.

Discussion

The delegate receives this message when a JavaScript function in the web view explicitly sets the status text. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also- [webViewStatusText:](#) (page 26)**Declared In**

WebUIDelegate.h

webView:setToolbarsVisible:

Sets whether a web view's toolbars should be visible.

- (void)webView:(WebView *)sender setToolbarsVisible:(BOOL)visible

Parameters*sender*

The web view that sent the message.

visible

If YES, all toolbars (with the exception of the status bar) are shown; otherwise, all toolbars (with the exception of the status bar) are removed.

Discussion

No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also- [webViewAreToolbarsVisible:](#) (page 21)**Declared In**

WebUIDelegate.h

webView:shouldPerformAction:fromSender:

Returns a Boolean value that indicates whether the action sent by the specified object should be performed.

- (BOOL)webView:(WebView *)sender shouldPerformAction:(SEL)action
fromSender:(id)fromObject**Parameters***sender*

The web view that sent the message.

*action*The action to perform. See *WebView Class Reference* for information on actions a web view can perform.*fromObject*

The object that sent the action.

Return Value

YES if the action should be performed; otherwise, NO.

DiscussionThis method allows the delegate to control the web view's behavior when action methods are invoked. For example, if the action is `copy:`, the delegate can return NO to perform a copy in some other way than the default.**Availability**

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webView:validateUserInterfaceItem:defaultValidation:

Returns a Boolean value that indicates whether the specified user interface item is valid.

```
- (BOOL)webView:(WebView *)sender validateUserInterfaceItem:(id <
    NSValidatedUserInterfaceItem >)item defaultValidation:(BOOL)defaultValidation
```

Parameters*sender*

The web view that sent the message.

item

The user interface item being validated.

defaultValidation

YES if the web view believes the user interface item is valid; otherwise, NO.

Return Value

YES if the specified user interface item is valid; otherwise, NO.

Discussion

See *NSUserInterfaceValidations Protocol Reference* and *NSValidatedUserInterfaceItem Protocol Reference* for more information about user interface validation. If you do not implement this method, the value of *defaultValidation* is used.

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webView:willPerformDragDestinationAction:forDraggingInfo:

Tells the receiver that the sending web view will perform the specified drag-destination action.

```
- (void)webView:(WebView *)sender
    willPerformDragDestinationAction:(WebDragDestinationAction)action
    forDraggingInfo:(id < NSDraggingInfo >)draggingInfo
```

Parameters*sender*

The web view that sent the message.

*action*The drag-destination action to perform. See [“Drag-Destination Actions”](#) (page 31) for a list of actions.*draggingInfo*

The information object for the dragging operation.

Discussion

This method is invoked after the last invocation of the [webView:dragDestinationActionMaskForDraggingInfo:](#) (page 10) method, when the dragged content is dropped and the sender is about to perform the destination action. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webView:willPerformDragSourceAction:fromPoint:withPasteboard:

Tells the receiver that the sending web view will perform the specified drag-source action.

```
- (void)webView:(WebView *)sender
    willPerformDragSourceAction:(WebDragSourceAction)action fromPoint:(NSPoint)point
    withPasteboard:(NSPasteboard *)pasteboard
```

Parameters

sender

The web view that sent the message.

action

The drag-source action to perform. See [“Drag-Source Actions”](#) (page 32) for a list of actions.

point

The point at which the drag began, specified in the coordinates of the web view.

pasteboard

The drag pasteboard.

Discussion

This method is invoked after the last invocation of the [webView:dragSourceActionMaskForPoint:](#) (page 11) method, when the dragged content is dropped and the sender is about to perform the drag-source action. The delegate has the opportunity to modify the contents of the object on the pasteboard before completing the drag-source action. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.3.9 and later.

Declared In

WebUIDelegate.h

webViewAreToolbarsVisible:

Returns a Boolean value indicating whether any toolbars are visible in a web view's window.

```
- (BOOL)webViewAreToolbarsVisible:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

YES if a web view's window has any toolbars that are currently visible (other than the status bar); otherwise, NO.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:setToolbarsVisible:](#) (page 18)

Declared In

WebUIDelegate.h

webViewClose:

Closes a web view in a window.

```
- (void)webViewClose:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Discussion

If you display multiple web views in a window then you might want to close only *sender* in your implementation. By default, this method sends the `close` method to the `NSWindow` object that contains *sender*.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

WebUIDelegate.h

webViewFirstResponder:

Returns the first responder of the web view's window.

```
- (NSResponder *)webViewFirstResponder:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

The view or subview that currently has the input focus. It can return `nil` or the default first responder if the sender is not attached to a window or if another view (not in the window) has the focus.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:makeFirstResponder:](#) (page 12)

Declared In

WebUIDelegate.h

webViewFocus:

Brings a web view's window to the front and makes it the active window.

```
- (void)webViewFocus:(WebView *)sender
```

Parameters*sender*

The web view that sent the message.

Discussion

By default, this method brings a web view's window into focus. If you display multiple web views in a window then you might also want to focus the input on *sender*, using [webView:makeFirstResponder:](#) (page 12).

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webViewUnfocus:](#) (page 27)

Declared In

WebUIDelegate.h

webViewFooterHeight:

Returns the height of the web view's printed page footer.

```
- (float)webViewFooterHeight:(WebView *)sender
```

Parameters*sender*

The web view that sent the message.

Return Value

The height of the web view's printed page footer. Returns 0.0 if no space is reserved for the footer.

Discussion

The height returned by this method is used to calculate the rectangle passed to the [webView:drawFooterInRect:](#) (page 11) method.

Availability

Available in Mac OS X v10.4.11 and later.

See Also

- [webView:drawFooterInRect:](#) (page 11)

Declared In

WebUIDelegate.h

webViewFrame:

Returns the frame rectangle of a web view's window.

```
- (NSRect)webViewFrame:(WebView *)sender
```

Parameters*sender*

The web view that sent the message.

Return Value

The frame rectangle of the web view's window.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:setFrame:](#) (page 16)

Declared In

WebUIDelegate.h

webViewHeaderHeight:

Returns the height of the web view's printed page header.

```
- (float)webViewHeaderHeight:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

The height of the web view's printed page header. Returns 0.0 if no space is reserved for the header.

Discussion

The height returned by this method is used to calculate the rectangle passed to the [webView:drawHeaderInRect:](#) (page 12) method.

Availability

Available in Mac OS X v10.4.11 and later.

See Also

- [webView:drawHeaderInRect:](#) (page 12)

Declared In

WebUIDelegate.h

webViewIsResizable:

Returns a Boolean value indicating whether a web view's window can be resized.

```
- (BOOL)webViewIsResizable:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

YES if the web view's window can be resized; otherwise, NO.

Discussion

If you display multiple web views in a window then your user interface delegate should implement this method to handle this special case.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:setResizable:](#) (page 17)

Declared In

WebUIDelegate.h

webViewIsStatusBarVisible:

Returns a Boolean value indicating whether the status bar in a web view's window is visible.

```
- (BOOL)webViewIsStatusBarVisible:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

YES if a web view's status bar (if any) is visible; otherwise, NO.

Discussion

If you do not implement this method, it returns NO by default.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:setStatusBarVisible:](#) (page 17)

Declared In

WebUIDelegate.h

webViewRunModal:

Displays a web view in a modal window.

```
- (void)webViewRunModal:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Discussion

This method should display and order front a modal window containing the specified web view. This method is invoked after the [webView:createWebViewModalDialogWithRequest:](#) (page 9) method is used to create a new window.

Availability

Available in Mac OS X v10.4.11 and later.

Declared In

WebUIDelegate.h

webViewShow:

Displays a web view's window and moves it to the front.

```
- (void)webViewShow:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Discussion

This method is typically used after a call to [webView:createWebViewWithRequest:](#) (page 10), which creates a new window. The new window is not ordered to the front (or even shown) unless you implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Declared In

WebUIDelegate.h

webViewStatusText:

Returns the current status message from a web view's window.

```
- (NSString *)webViewStatusText:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Return Value

The status message displayed in the web view's window if one has been set with the [webView:setStatusText:](#) (page 18) method; otherwise, `nil`.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webView:setStatusText:](#) (page 18)

Declared In

WebUIDelegate.h

webViewUnfocus:

Relinquishes focus on a web view's window.

```
- (void)webViewUnfocus:(WebView *)sender
```

Parameters

sender

The web view that sent the message.

Discussion

This method releases focus for the entire window. If you display multiple web views in a window, you might instead want to change the input focus to another view, using the [webView:makeFirstResponder:](#) (page 12) method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

See Also

- [webViewFocus:](#) (page 22)

Declared In

WebUIDelegate.h

Constants

Menu Item Tags

Tags that define the types of default menu items passed to the [webView:contextMenuItemsForElement:defaultMenuItems:](#) (page 8) method.

```
enum {
    WebMenuItemTagOpenLinkInNewWindow = 1,
    WebMenuItemTagDownloadLinkToDisk,
    WebMenuItemTagCopyLinkToClipboard,
    WebMenuItemTagOpenImageInNewWindow,
    WebMenuItemTagDownloadImageToDisk,
    WebMenuItemTagCopyImageToClipboard,
    WebMenuItemTagOpenFrameInNewWindow,
    WebMenuItemTagCopy,
    WebMenuItemTagGoBack,
    WebMenuItemTagGoForward,
    WebMenuItemTagStop,
    WebMenuItemTagReload,
    WebMenuItemTagCut,
    WebMenuItemTagPaste,
    WebMenuItemTagSpellingGuess,
    WebMenuItemTagNoGuessesFound,
    WebMenuItemTagIgnoreSpelling,
    WebMenuItemTagLearnSpelling,
    WebMenuItemTagOther,
    WebMenuItemTagSearchInSpotlight,
    WebMenuItemTagSearchWeb,
    WebMenuItemTagLookUpInDictionary,
    WebMenuItemTagOpenWithDefaultApplication,
    WebMenuItemPDFActualSize,
    WebMenuItemPDFZoomIn,
    WebMenuItemPDFZoomOut,
    WebMenuItemPDFAutoSize,
    WebMenuItemPDFSinglePage,
    WebMenuItemPDFFacingPages,
    WebMenuItemPDFContinuous,
    WebMenuItemPDFNextPage,
    WebMenuItemPDFPreviousPage,
};
```

Constants

`WebMenuItemTagOpenLinkInNewWindow`

Open the link in a new window.

Available in Mac OS X v10.2 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagDownloadLinkToDisk`

Download the link to a disk.

Available in Mac OS X v10.2 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagCopyLinkToClipboard`

Copy the link to the clipboard.

Available in Mac OS X v10.2 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagOpenImageInNewWindow`

Open the image in a new window.

Available in Mac OS X v10.2 and later.

Declared in `WebUIDelegate.h`.

WebMenuItemTagDownloadImageToDisk

Download the image to disk.

Available in Mac OS X v10.2 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagCopyImageToClipboard

Copy the image to the clipboard.

Available in Mac OS X v10.2 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagOpenFrameInNewWindow

Open the frame in a new window.

Available in Mac OS X v10.2 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagCopy

Copy the element to the clipboard.

Available in Mac OS X v10.2 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagGoBack

Load the previous page.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagGoForward

Load the next page.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagStop

Stop loading the current page.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagReload

Reload the current page.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagCut

Cut the currently selected content.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

WebMenuItemTagPaste

Paste the content on the clipboard onto the current selection.

Available in Mac OS X v10.3 and later.

Declared in WebUIDelegate.h.

`WebMenuItemTagSpellingGuess`

Suggest spellings for the misspelled word.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagNoGuessesFound`

Indicate whether any suggested spellings for the misspelled word could be found.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagIgnoreSpelling`

Ignore the misspelled word.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagLearnSpelling`

Add the misspelled word to the user's list of acceptable words.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagOther`

Used when a tag for an item in the context menu can't be determined.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagSearchInSpotlight`

Search Spotlight for the current selection.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagSearchWeb`

Search the web for the current selection.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagLookUpInDictionary`

Look up the current selection in the Dictionary.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemTagOpenWithDefaultApplication`

Open the current selection using the default application.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFActualSize`

Display a PDF document at its original size.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFZoomIn`

Scale up a PDF document.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFZoomOut`

Scale down a PDF document.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFAutoSize`

Display a PDF document at a user-specified size.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFSinglePage`

Display a PDF document one page at a time.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFFacingPages`

Display a PDF document two pages at a time.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFContinuous`

Display all pages in a PDF document continuously, using a vertical scroll bar, if necessary.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFNextPage`

Display the next page of a PDF document.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

`WebMenuItemPDFPreviousPage`

Display the previous page of a PDF document.

Available in Mac OS X v10.4.11 and later.

Declared in `WebUIDelegate.h`.

Discussion

These tags define common actions a user might want to take with elements in a page. You can use the tags to differentiate between the different types of menu items.

Declared In

`WebUIDelegate.h`

Drag-Destination Actions

Actions that the destination object of a drag operation can perform.

```
typedef enum {
    WebDragDestinationActionNone    = 0,
    WebDragDestinationActionDHTML  = 1,
    WebDragDestinationActionEdit   = 2,
    WebDragDestinationActionLoad   = 4,
    WebDragDestinationActionAny    = UINT_MAX
} WebDragDestinationAction;
```

Constants

`WebDragDestinationActionNone`

No action.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragDestinationActionDHTML`

Allows DHTML (such as JavaScript) to handle the drag.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragDestinationActionEdit`

Allows editable documents to be changed by the drag operation.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragDestinationActionLoad`

Allows the drag operation to change the location.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragDestinationActionAny`

Allows any defined action to occur.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

Declared In

`WebUIDelegate.h`

Drag-Source Actions

Actions that the source object of a drag operation can perform.


```
typedef enum {
    WebDragSourceActionNone           = 0,
    WebDragSourceActionDHTML         = 1,
    WebDragSourceActionImage         = 2,
    WebDragSourceActionLink          = 4,
    WebDragSourceActionSelection     = 8,
    WebDragSourceActionAny           = UINT_MAX
} WebDragSourceAction;
```

Constants

`WebDragSourceActionNone`

No action.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragSourceActionDHTML`

Allows DHTML (such as JavaScript) in the source object to initiate a drag operation.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragSourceActionImage`

Allows the user to drag an image in the source object.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragSourceActionLink`

Allows the user to drag a link in the source object.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragSourceActionSelection`

Allows the user to drag a selection in the source object.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

`WebDragSourceActionAny`

Allows any defined action to occur.

Available in Mac OS X v10.3 and later.

Declared in `WebUIDelegate.h`.

Deprecated WebUIDelegate Methods

A method identified as deprecated has been superseded and may become unsupported in the future.

Deprecated in Mac OS X v10.4.11

webView:runJavaScriptAlertPanelWithMessage:

Displays a JavaScript alert panel. (Deprecated in Mac OS X v10.4.11. Use [webView:runJavaScriptAlertPanelWithMessage:initiatedByFrame:](#) (page 14) instead.)

```
- (void)webView:(WebView *)sender runJavaScriptAlertPanelWithMessage:(NSString *)message
```

Parameters

sender

The web view that sent the message.

message

The message to display in the alert panel.

Discussion

This method is used to display a panel when JavaScript code calls `alert`. Delegates should visually indicate that this panel comes from JavaScript. The panel should have, for example, a single OK button. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Deprecated in Mac OS X v10.4.11.

See Also

- [webView:runJavaScriptConfirmPanelWithMessage:](#) (page 35)
- [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:](#) (page 36)

Declared In

WebUIDelegate.h

webView:runJavaScriptConfirmPanelWithMessage:

Displays a JavaScript confirm panel. (Deprecated in Mac OS X v10.4.11. Use [webView:runJavaScriptConfirmPanelWithMessage:initiatedByFrame:](#) (page 15) instead.)

```
- (BOOL)webView:(WebView *)sender runJavaScriptConfirmPanelWithMessage:(NSString *)message
```

Deprecated WebUIDelegate Methods

Parameters*sender*

The web view that sent the message.

message

The message to display in the confirmation panel.

Discussion

This method is used to display a confirmation panel when JavaScript code calls `confirm`. It returns `YES` if confirmed, `NO` otherwise. Delegates should visually indicate that this panel comes from JavaScript. The panel should have, for example, an OK and Cancel button. No action is taken if you do not implement this method.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Deprecated in Mac OS X v10.4.11.

See Also

- [webView:runJavaScriptAlertPanelWithMessage:](#) (page 35)
- [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:](#) (page 36)

Declared In

WebUIDelegate.h

webView:runJavaScriptTextInputPanelWithPrompt:defaultText:

Displays a JavaScript text input panel and returns the entered text. (Deprecated in Mac OS X v10.4.11. Use [webView:runJavaScriptTextInputPanelWithPrompt:defaultText:initiatedByFrame:](#) (page 15) instead.)

```
- (NSString *)webView:(WebView *)sender
  runJavaScriptTextInputPanelWithPrompt:(NSString *)prompt defaultText:(NSString *)defaultText
```

Parameters*sender*

The web view that sent the message.

prompt

The message to display in the text input panel.

defaultText

Default placeholder text to display in the text field.

Discussion

This method is used to provide an alternative prompt panel when JavaScript code calls `prompt`. Delegates should visually indicate that this panel comes from JavaScript. The panel should have an OK and Cancel button, and an editable text field. If you do not implement this method, a JavaScript text input panel is displayed.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Deprecated in Mac OS X v10.4.11.

See Also

- [webView:runJavaScriptAlertPanelWithMessage:](#) (page 35)
- [webView:runJavaScriptConfirmPanelWithMessage:](#) (page 35)

Declared In

WebUIDelegate.h

webView:setContentRect:

Sets the window's content view frame to the specified content rectangle. (Deprecated in Mac OS X v10.4.11. Content rectangle calculations are automatic.)

```
- (void)webView:(WebView *)sender setContentRect:(NSRect)contentRect
```

Parameters*sender*

The web view that sent the message.

contentRect

The location and size of the window's content area.

Discussion

The content view is the highest accessible `NSView` object in the view hierarchy displayed in the window. A web view invokes this method instead of setting the content view's frame directly, allowing delegates to augment the behavior by, for example, avoiding auto-saving of the size.

If this method is not implemented by the delegate, then [webView:setFrame:](#) (page 16) is invoked with the rectangle returned by sending the `NSWindow` method `frameRectForContentRect:styleMask:` to the window.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Deprecated in Mac OS X v10.4.11.

See Also

- [webViewContentRect:](#) (page 37)

Declared In

WebUIDelegate.h

webViewContentRect:

Returns a web view window's content rectangle. (Deprecated in Mac OS X v10.4.11. Content rectangle calculations are automatic.)

```
- (NSRect)webViewContentRect:(WebView *)sender
```

Parameters*sender*

The web view that sent the message.

Deprecated WebUIDelegate Methods

Return Value

The content rectangle of the window that contains the web view.

Discussion

The content view is the highest accessible `NSView` object in the view hierarchy displayed in the window. A web view invokes this method instead of setting the content view's frame directly, allowing delegates to alter the size that is returned.

Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

Deprecated in Mac OS X v10.4.11.

See Also

- [webView:setContentRect:](#) (page 37)

Declared In

WebUIDelegate.h

Document Revision History

This table describes the changes to *WebUIDelegate Protocol Reference*.

Date	Notes
2009-03-04	Added separate parameter and return-value descriptions.
2008-10-15	Minor edits throughout.
2008-06-19	Updated for Mac OS X v10.5.
2006-07-31	Includes new API for Mac OS X v.10.5
2006-05-23	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

D

Drag-Destination Actions [31](#)
Drag-Source Actions [32](#)

M

Menu Item Tags [27](#)

W

WebDragDestinationActionAny [constant 32](#)
WebDragDestinationActionDHTML [constant 32](#)
WebDragDestinationActionEdit [constant 32](#)
WebDragDestinationActionLoad [constant 32](#)
WebDragDestinationActionNone [constant 32](#)
WebDragSourceActionAny [constant 33](#)
WebDragSourceActionDHTML [constant 33](#)
WebDragSourceActionImage [constant 33](#)
WebDragSourceActionLink [constant 33](#)
WebDragSourceActionNone [constant 33](#)
WebDragSourceActionSelection [constant 33](#)
WebMenuItemPDFActualSize [constant 30](#)
WebMenuItemPDFAutoSize [constant 31](#)
WebMenuItemPDFContinuous [constant 31](#)
WebMenuItemPDFFacingPages [constant 31](#)
WebMenuItemPDFNextPage [constant 31](#)
WebMenuItemPDFPreviousPage [constant 31](#)
WebMenuItemPDFSinglePage [constant 31](#)
WebMenuItemPDFZoomIn [constant 31](#)
WebMenuItemPDFZoomOut [constant 31](#)
WebMenuItemTagCopy [constant 29](#)
WebMenuItemTagCopyImageToClipboard [constant 29](#)
WebMenuItemTagCopyLinkToClipboard [constant 28](#)
WebMenuItemTagCut [constant 29](#)
WebMenuItemTagDownloadImageToDisk [constant 29](#)
WebMenuItemTagDownloadLinkToDisk [constant 28](#)
WebMenuItemTagGoBack [constant 29](#)

WebMenuItemTagGoForward [constant 29](#)
WebMenuItemTagIgnoreSpelling [constant 30](#)
WebMenuItemTagLearnSpelling [constant 30](#)
WebMenuItemTagLookUpInDictionary [constant 30](#)
WebMenuItemTagNoGuessesFound [constant 30](#)
WebMenuItemTagOpenFrameInNewWindow [constant 29](#)
WebMenuItemTagOpenImageInNewWindow [constant 28](#)
WebMenuItemTagOpenLinkInNewWindow [constant 28](#)
WebMenuItemTagOpenWithDefaultApplication
[constant 30](#)
WebMenuItemTagOther [constant 30](#)
WebMenuItemTagPaste [constant 29](#)
WebMenuItemTagReload [constant 29](#)
WebMenuItemTagSearchInSpotlight [constant 30](#)
WebMenuItemTagSearchWeb [constant 30](#)
WebMenuItemTagSpellingGuess [constant 30](#)
WebMenuItemTagStop [constant 29](#)
webViewAreToolbarsVisible: <NSObject> instance
[method 21](#)
webViewClose: <NSObject> instance [method 22](#)
webView:contextMenuItemsForElement:
defaultMenuItems: <NSObject> instance [method 8](#)
webView:createWebViewModalDialogWithRequest:
protocol instance [method 9](#)
webView:createWebViewWithRequest: <NSObject>
instance [method 10](#)
webView:dragDestinationActionMaskForDraggingInfo:
<NSObject> instance [method 10](#)
webView:dragSourceActionMaskForPoint:
<NSObject> instance [method 11](#)
webView:drawFooterInRect: protocol instance
[method 11](#)
webView:drawHeaderInRect: protocol instance
[method 12](#)
webView:makeFirstResponder: <NSObject> instance
[method 12](#)
webView:mouseDidMoveOverElement:modifierFlags:
<NSObject> instance [method 13](#)
webView:printFrameView: protocol instance [method 13](#)

- webView:runBeforeUnloadConfirmPanelWithMessage:
initiatedByFrame: <NSObject> instance method
14
- webView:runJavaScriptAlertPanelWithMessage:
<NSObject> instance method 35
- webView:runJavaScriptAlertPanelWithMessage:
initiatedByFrame: <NSObject> instance method
14
- webView:runJavaScriptConfirmPanelWithMessage:
<NSObject> instance method 35
- webView:runJavaScriptConfirmPanelWithMessage:
initiatedByFrame: <NSObject> instance method
15
- webView:runJavaScriptTextInputPanelWithPrompt:
defaultText: <NSObject> instance method 36
- webView:runJavaScriptTextInputPanelWithPrompt:
defaultText:initiatedByFrame: <NSObject>
instance method 15
- webView:
runOpenPanelForFileButtonWithResultListener:
<NSObject> instance method 16
- webView:setContentRect: <NSObject> instance
method 37
- webView:setFrame: <NSObject> instance method 16
- webView:setResizable: <NSObject> instance method
17
- webView:setStatusBarVisible: <NSObject> instance
method 17
- webView:setStatusText: <NSObject> instance
method 18
- webView:setToolbarsVisible: <NSObject> instance
method 18
- webView:shouldPerformAction:fromSender:
<NSObject> instance method 19
- webView:validateUserInterfaceItem:
defaultValidation: <NSObject> instance method
19
- webView:willPerformDragDestinationAction:
forDraggingInfo: <NSObject> instance method
20
- webView:willPerformDragSourceAction:fromPoint:
withPasteboard: <NSObject> instance method
21
- webViewContentRect: <NSObject> instance method
37
- webViewFirstResponder: <NSObject> instance
method 22
- webViewFocus: <NSObject> instance method 22
- webViewFooterHeight: protocol instance method 23
- webViewFrame: <NSObject> instance method 23
- webViewHeaderHeight: protocol instance method 24
- webViewIsResizable: <NSObject> instance method
24
- webViewIsStatusBarVisible: <NSObject> instance
method 25
- webViewRunModal: protocol instance method 25
- webViewShow: <NSObject> instance method 26
- webViewStatusText: <NSObject> instance method 26
- webViewUnfocus: <NSObject> instance method 27