

spellingPanel

setAccessoryView:
accessoryView
setWordFieldValue:

Setting the language setLanguage

language:

Checking spelling checkSpelling:of:

checkSpelling:of:wordCount:

Managing ignored words setIgnoredWordsForSpellDocument:

ignoredWords:forSpellDocument:

accessoryView

Returns the customized View of the Spelling panel previously established by setAccessoryView:.

setAccessoryView:

(BOOL)checkSpelling:(NXSpellCheckMode)mode
of:(id <NXReadOnlyTextStream, NXSelectRange>)anObject

The first argument mode indicates the mode in which the spelling checker will operate, as defined in NXSpellCheckMode. Values that can be used with checkSpelling:of: are as follows:

NX_CheckSpellingChecks spelling of the entire text stream, in sequence from the current character offset to the end, then from the start to the current character offset.

NX_CheckSpellingFromStartChecks spelling of the entire text stream, from start to end.

NX_CheckSpellingToEndChecks spelling from the current character offset to the end of the text stream.

NX_CheckSpellingInSelectionChecks spelling in the selected portion of the text stream.

The method returns YES when the check has found a misspelled word, NO when it has searched the entire text without finding one.

checkSpelling:of:wordCount:, currentCharacterOffset (NXReadOnlyTextStream protocol).

(BOOL)checkSpelling:(NXSpellCheckMode)mode
of:(id <NXReadOnlyTextStream, NXSelectRange>)anObject
wordCount:(int *)theCount

This method looks for a misspelled word and at the same time counts the number of words scanned. mode of search or count, a member of the enumeration type NXSpellCheckMode. Values that can be used with checkSpelling:of:wordCount: are as follows:

NX_CheckSpellingChecks spelling and counts words for the entire text stream, in sequence from the current character offset to the end, then from the beginning to the current character offset.

NX_CheckSpellingFromStartChecks spelling and counts words for the entire text stream, from start to end.

NX_CheckSpellingToEndChecks spelling and counts words from the current character offset to the end of the text stream.

NX_CheckSpellingInSelectionChecks spelling and counts words in the selected portion of the text stream.

NX_CountWordsCounts the number of words in the entire text stream doesn't check spelling.

checkSpelling:of:, currentCharacterOffset (NXReadOnlyTextStream protocol)

closeSpellDocument:(int)tag

Notifies the NXSpellChecker object that the user has finished with the document identified by tag. The application can then discard the temporary list of ignorable words that it compiled for this document. The argument identifies the document by a numeric tag, previously returned by spellDocumentTag. Returns self.

spellDocumentTag (NXIgnoreMisspelledWords protocol)

(char **)ignoredWordsForSpellDocument:(int)tag

Returns the list of ignored words that the spell checker has accumulated for the document identified by tag. The application may want to preserve the list of ignored words so that they can be ignored in future documents. To preserve the ignored word list, it should invoke this method after spelling has been completed and sends a closeSpellDocument: message. The argument identifies the document by the tag that spellDocumentTag previously returned.

spellDocumentTag (NXIgnoreMisspelledWords protocol)

(const char *)language

Returns the character string that identifies the English name of the currently selected language. If the application temporarily overrides the current language (by invoking setLanguage:), this method will be useful to retrieve the language so that it can subsequently be restored. Otherwise, the application will not ordinarily need to call this method.

setLanguage

setAccessoryView:aView

Adds aView to the contents of the Spelling panel. An application can invoke this method to add custom views to the panel's functions. The Spelling panel is automatically resized to accommodate aView. This method can be invoked repeatedly to change the accessory View depending on the situation. When aView is nil, the effect is to restore the accessory View that's already in the panel. Returns the former accessory view, or nil if there was none.

setIgnoredWords:(const char *const *)someWords forSpellDocument:(int)tag

Initializes the NXSpellChecker's list of acceptable words for the document identified by tag. The first argument is an array of pointers to the words. The second argument identifies the document for which the list is being set (described above in the section "Matching a List of Ignored Words with the Document It Belongs To").

spellDocumentTag (NXIgnoreMisspelledWords protocol)

words. Suppose that `checker` is the id of an `NXSpellChecker` instance, and the application sends messages:

`systemLanguages (Application)`

`spellingPanel`

Returns the Panel in which the user can choose a dictionary or select actions with respect to misspellings. You need to identify the Panel in order to bring it forward when the user selects `Spelling` in the application. You may also need it if your application elects to modify the panel with `setAccessoryView`:

`setAccessoryView:`