

MiscStringArray

Inherits From: Object
Declared In: <misckit/MiscStringArray.h>

Class Description

A MiscStringArray keeps track of a list of MiscString objects (or subclasses thereof). Upon request, it can build up an array of constant string pointers (const char **). The array of pointers can be made into all unique strings, if desired.

A MiscStringArray is created as usual via +**alloc** and ±**init**. Strings may be added to the MiscStringArray via the ±**addString:** and ±**insertString:at:** methods. A String may be removed with the ±**removeString:** method. A string may be searched for with the ±**stringAt:** and ±**indexOfString:** methods. Use ±**count** or ±**stringCount** to obtain the number of strings stored in the MiscStringArray. You can use the array as a delegate to the NXBrowser class, filling it with all the strings in the array.

To directly manipulate the List object which contains the respective MiscStrings, use the ±**strings** method. To obtain the actual char ** pointer to the array of strings, use the ±**stringArray** method. If the array returned should contain unique strings, then first send the ±**setUniqued:** method before requesting the string array. You can check to see if the returned string array is all unique strings with the ±**uniqued** method.

This implementation is rather bare-bones and will be enhanced in future MiscKit releases to make it more useful.

Instance Variables

```
List *strings;  
const char **stringArray;  
BOOL uniqued;
```

strings

A List object containing MiscStrings, used to store the actual data.

stringArray

A pointer to the string array. NULL if the array has not yet been built.

uniqued

Whether or not the string array should be built with unique strings.

Method Types

Adding Strings

- addString:
- insertString:at:

Querying/Changing Attributes

- count
- setUniqued:
- stringCount
- uniqued

Retreiving Strings

- indexOfString:
- stringArray
- stringAt:
- strings

AppKit support

- browser:fillMatrix:inColumn:

Instance Methods

addString:

- **addString:**(const char *)*aString*

Adds *aString* to the MiscStringArray. Returns *self*.

See also: **±insertString:At:** and **±removeString:**

browser:fillMatrix:inColumn:

- (int)**browser:**sender **fillMatrix:**matrix **inColumn:**(int)column

NXBrowser delegate method. This method fills up the browser which sent the message with the strings contained by the MiscString Array. Returns the number of strings contained by the receiver.

count

- (unsigned int)**count**

Returns the number of strings stored in the MiscStringArray.

See also: **±stringCount**

indexOfString:

- (unsigned int)**indexOfString:**(const char *)*aString*

Returns the position of the string *aString* in the MiscStringArray, from zero to one less than the number of stored strings.

insertString:at:

- **insertString:**(const char *)*aString*
at:(unsigned int)*index*

Insert *aString* into the MiscStringArray before the string at *index*. Return *self*.

See also: `±addString:` and `±removeString:`

removeString:

- `removeString:(const char *)aString`

Removes *aString* from the MiscStringArray. Returns *self* if the string was in the MiscStringArray and *nil* if it wasn't found in the array.

See also: `±addString:`

setUniqued:

- `setUniqued:(BOOL)yn`

Sets wheter or not the array returned by *stringArray* should contain all uniqued strings. Return *self*.

See also: `±uniqued`

stringArray

- `(const char **)stringArray`

Returns an array of the strings stored in the MiscStringArray object. If the unique flag has been set, all the strings will be uniqued before the pointer is returned.

See also: `±strings`

stringAt:

- `(const char *)stringAt:(unsigned int)index`

Returns a pointer to the string stored at *index* in the MiscStringArray.

stringCount

- (unsigned int)**stringCount**

Returns the number of strings stored in the MiscStringArray.

See also: `±count`

strings

- **strings**

Returns a List object containing MiscStrings. This List is used to generate the string array and is owned by the MiscStringArray which returns it. Therefore, you should not free it, although you are free to modify the list.

See also: `±stringArray`

uniqued

- (BOOL)**uniqued**

Returns whether or not all the strings in the string array will be made unique.

See also: `±setUniqued:`