

Application Additions

Inherits From:	Responder : Object
Declared In:	apps/Preferences.h

Category Description

Preferences.h declares a category that adds four methods to the Application class of the Application Kit. These methods make it easier for your Preferences module to:

- Locate its interface when the module is loaded
- Enable and disable items in the Windows and Edit menus of the Preferences application
- Access the views contained in the Preferences window

Method Types

Loading the interface	<code>-loadNibForLayout:owner:</code>
Controlling menu items	<code>-enableEdit:</code> <code>-enableWindow:</code>
Accessing the Preferences window	<code>-appWindow</code>

Instance Methods

appWindow

`-appWindow`

Returns the **id** of the Preferences window, enabling you to alter its content view, for example.

enableEdit:

`-enableEdit:(int)aMask`

Enables and disables menu items in Preferences' Edit menu. *aMask* specifies which items are to be enabled. For example, this message enables the Cut and Copy commands:

```
[NXApp enableEdit: CUT_ITEM|COPY_ITEM];
```

The permitted values for *aMask* are:

```
CUT_ITEM
COPY_ITEM
PASTE_ITEM
SELECTALL_ITEM
EDIT_ALL_ITEMS
```

See also: `-enableWindow:`

enableWindow:

-[enableWindow:](int)*aMask*

Enables and disables menu items in Preferences' Window menu. *aMask* specifies which items are to be enabled. The permitted values for *aMask* are:

MINIATURIZE_ITEM
CLOSE_ITEM
WINDOW_ALL_ITEMS

See also: **-[enableEdit:]**

loadNibForLayout:owner:

-[loadNibForLayout:](const char *)*name* **owner:***anOwner*

Loads the nib file named *"name.nib"* and makes *anOwner* its owner.

This is a convenience method that searches for the nib file in the appropriate language subproject of the bundle from which the class of *anOwner* was loaded.

See also: **-[bundleForClass:]** (NSBundle common class)