```
(BOOL)printDialog; // Print a document.
- (void) print:(id)x withProperties:(NSDictionary *)withProperties printDialog:
- (id) open:(id)x; // Open a document.
@property (copy, readonly) NSString *version; // The version number of the application.
@property (readonly) BOOL frontmost; // Is this the active application?
@property (copy, readonly) NSString *name; // The name of the application.
- (SBElementArray<SystemSettingsWindow *> *) windows;
- (SBElementArray<SystemSettingsDocument *> *) documents;
@interface SystemSettingsApplication : SBApplication
// The application's top-level scripting object.
 * Standard Suite
\*
@end
(BOOL)printDialog; // Print a document.
- (void) printWithProperties: (NSDictionary *)withProperties printDialog:
Close a document.
- (void) closeSaving:(SystemSettingsSaveOptions)saving savingIn:(NSURL *)savingIn; //
@protocol SystemSettingsGenericMethods
typedef enum SystemSettingsPrintingErrorHandling SystemSettingsPrintingErrorHandling;
};
PostScript errors */
    SystemSettingsPrintingErrorHandlingDetailed = 'lwdt' /* print a detailed report of
hand \lim */,
    SystemSettingsPrintingErrorHandlingStandard = 'lwst' /* Standard PostScript error
enum SystemSettingsPrintingErrorHandling {
typedef enum SystemSettingsSaveOptions SystemSettingsSaveOptions;
};
file. */
    SystemSettingsSaveOptionsAsk = 'ask ' /* Ask the user whether or not to save the
    SystemSettingsSaveOptionsNo = 'no ' /* Do not save the file. */,
    SystemSettingsSaveOptionsYes = 'yes ' /* Save the file. */,
enum SystemSettingsSaveOptions {
SystemSettingsPane, SystemSettingsAnchor;
@class SystemSettingsApplication, SystemSettingsDocument, SystemSettingsWindow,
#import <ScriptingBridge/ScriptingBridge.h>
#import <AppKit/AppKit.h>
 *\
 * SystemSettings.h
\*
```

```
    (void) quitSaving:(SystemSettingsSaveOptions)saving; // Quit the application.
    (ROOL) exists:(id)x: // Vosify that application.
```

@property (readonly) BOOL modified; // Has it been modified since the last save?
@property (copy, readonly) NSURL *file; // Its location on disk, if it has one.

@interface SystemSettingsDocument : SBObject <SystemSettingsGenericMethods>

- (BOOL) exists:(id)x; // Verify that an object exists.

@property (copy, readonly) NSString *name; // Its name.

@end

// A document.

@end

true.

window.

```
- (SBElementArray<SystemSettingsPane *> *) panes;
@interface SystemSettingsApplication (SystemSettings)
// The System Settings top-level scripting object.
*\
 * System Settings
@end
contents are displayed in the window.
@property (copy, readonly) SystemSettingsDocument *document; // The document whose
@property BOOL zoomed; // Is the window zoomed right now?
@property (readonly) BOOL zoomable; // Does the window have a zoom button?
@property BOOL visible; // Is the window visible right now?
@property (readonly) BOOL resizable; // Can the window be resized?
@property BOOL miniaturized; // Is the window minimized right now?
@property (readonly) BOOL miniaturizable; // Does the window have a minimize button?
@property (readonly) BOOL closeable; // Does the window have a close button?
@property NSRect bounds; // The bounding rectangle of the window.
@property NSInteger index; // The index of the window, ordered front to back.
- (NSInteger) id; // The unique identifier of the window.
@property (copy, readonly) NSString *name; // The title of the window.
@interface SystemSettingsWindow : SBObject <SystemSettingsGenericMethods>
// A window.
@end
```

@property BOOL showAll; // Is System Settings in show—all view? (Setting to false does nothing.) Deprecated: setting this property no longer does anything; it is always set to

@property (copy) SystemSettingsPane *currentPane; // The currently selected pane.
@property (copy, readonly) SystemSettingsWindow *settingsWindow; // The main settings

// A settings pane.
@interface SystemSettingsPane : SBObject <SystemSettingsGenericMethods>

- (SBElementArray<SystemSettingsAnchor *> *) anchors;
- (NSString *) id; // The id of the settings pane. @property (copy, readonly) NSString *name; // The name of the settings pane.
- (SystemSettingsPane *) authorize; // Prompt for authorization for a settings pane.Deprecated: no longer does anything.

@end

// An anchor within a settings pane.
@interface SystemSettingsAnchor : SBObject <SystemSettingsGenericMethods>

- (id) reveal; // Reveals a settings pane or an anchor within a pane.

@property (copy, readonly) NSString *name; // The name of the anchor.

@end