Enumerations

Name	Values
	This class exposes no enumerations.

Constants

Name	Type	Value
		This class exposes no constants.

Events

Definition	Description
	This class exposes no events.

Methods

Name	Parameters	Return Type	Description
AddPreset	name as String values as Dictionary	None	Adds a new static set of preset values to the instance.
Erase	DeleteFile as Boolean = False	None	Clears all settings. If DeleteFile is True, then the settings file is removed from the filesystem.
GetValue	strKey as String Optional DefaultValue as String = “”	Variant	Retrieves a value for the specified setting. DefaultValue will be returned if no entry is found for strKey.
HasPreset	name as String	Boolean	Returns True if a preset has been added with the name specified by the parameter.
Load	None	None	Reload contents from settings file.
LoadPreset	name as String ignoredValues() as String	None	Loads the specified preset in to the main settings memory, overwrites any value whose name does not appear in the ignoredValues array.
Preset	name as String	Dictionary	Returns the dictionary represented by the specified preset name.
RemoveSetting	strKey as String	None	Remove a setting from the instance and settings file.
RemoveAllPresets	None	None	Clears all added presets.
RemovePreset	name as String	None	Removes the specified preset.
Save	None	None	Update settings file with instance values.
SetValue	strKey as String Value as Variant	None	Sets a new setting in the instance with value supplied by Value.

Lookups

Definition	Description
(Key as String, Assigns Value as String)	Sets the value of a setting by key.
(Key as String, DefaultValue as String = “”)	Returns the currently stored value for a key or DefaultValue if not set.

Properties

Name	Type	Default Value	Description
AlwaysSave	Boolean	True	If True, any change to the instance's settings will save all values to the file immediately.
Values	Dictionary	Nil	The raw dictionary containing all setting values.

Constructor

```
Constructor()
```

Will automatically select the best place and filename for your settings file. On MacOS the filename is set to the Bundle Identifier, on Linux and Windows it is set to the executable's filename. On all OSes it is saved to SpecialFolder.Preferences.

```
Constructor( fileName as String )
```

Uses a settings file within SpecialFolder.Preferences whose name is contained within the fileName parameter.

```
Constructor( settingsPath as FolderItem )
```

Uses the specified file, anywhere on the system.

Examples

Advanced Usage

GraffitiSettings makes no assumptions about your data, and largely expects that values supplied will be readily converted to strings for storage in XML. If you are using things like Arrays, Pictures, etc that don't readily convert to string values, then you should make those conversions yourself before setting the value or after getting the value. As an example, if you wish to store an array of strings, you should use the `Join()` and `Split()` methods in the Xojo framework to prepare your values.

Operator Lookup

Operator Lookup allows you to use keys directly as though they were properties of [illegible] object. To set values you can implement this as:

```
mySettings.TextFont = "System"
```

Or to retrieve a value:

```
dim myValue as String = mySettings.TextFont("System")
```

Changes to this Page

Date	Edit	Editor
01/29/2018	Document Created	