

in

COLLABORATORS

	TITLE : in		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		July 31, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	in	1
1.1	stringclass.guide	1
1.2	stringclass/--background--	1
1.3	stringclass/SM_FORMAT_STRING	2
1.4	stringclass/STRINGA_Integer[xxx]	2
1.5	stringclass/STRINGA_MinCharsVisible	3
1.6	stringclass/STRINGA_StringInfo	3

Chapter 1

in

1.1 stringclass.guide

```
Search
TABLE OF CONTENTS

stringclass/--background--
stringclass/SM_FORMAT_STRING
stringclass/STRINGA_Integer[xxx]
stringclass/STRINGA_MinCharsVisible
stringclass/STRINGA_StringInfo
```

1.2 stringclass/--background--

NAME

```
Class:      stringclass
Superclass: baseclass
Include File: <libraries/bgui.h>
```

FUNCTION

To provide a gadget similar to the gadtools.library string and integer gadget kinds. Objects of this class send out the following attribute pairs in their notification events:

```
GA_ID - Gadget object ID.
STRINGA_TextVal - Gadget object string contents (string objects).
STRINGA_LongVal - Gadget object integer contents (integer objects).
```

NOTE

All system strgclass attributes are also supported by this class. Please refer to the documentation on the strgclass for more information on these attributes.

1.3 stringclass/SM_FORMAT_STRING

NAME

SM_FORMAT_STRING -- Format object contents.

SYNOPSIS

```
err = DoMethod( obj, SM_FORMAT_STRING, gi, fstr, ... )
```

```
ULONG  err;  
struct GadgetInfo *gi;  
UBYTE *fstr;
```

FUNCTION

This method may be used to set a formatted string in a string object. The formatted string may also contain any locale.library compatible formatting codes.

INPUTS

gi - This must point to a valid GadgetInfo structure if the change must also be shown visually on screen. The BGUI_DoGadgetMethod() call will automatically construct this for you.

fstr - This must point to a 0-terminated string which contains the formatting codes. Also all locale.library FormatString() formatting codes are supported when the locale.library is available.

Following the format string you can put the formatting arguments which may be needed to result in a correct output.

NOTE

This method will only work on string objects, and not on integer objects.

RESULT

err - TRUE upon success and FALSE upon failure.

SEE ALSO

bgui.library/BGUI_DoGadgetMethodA()

1.4 stringclass/STRINGA_Integer[xxx]

NAME

STRINGA_IntegerMin, STRINGA_IntegerMax -- (ULONG) ** V39 **

FUNCTION

To set the minimum and maximum values which the integer object can hold. When a `STRINGA_LongVal` is set which is out of this range the value is adjusted to be inside these boundaries. Also when the user enters a value outside these boundaries the screen will flash and the entered value is adjusted and not accepted.

DEFAULTS

0xEFFFFFFF minimum and 0xFFFFFFFF maximum.

APPLICABILITY

(ISU) .

1.5 stringclass/STRINGA_MinCharsVisible

NAME

`STRINGA_MinCharsVisible` -- (ULONG) ** V39 **

FUNCTION

To specify the minimum number of characters which should always stay visible. This number is involved in calculating the object's minimum size. The object won't resize any smaller if that would mean that the space required to show this number of characters would become insufficient.

DEFAULT

2.

APPLICABILITY

(I) .

1.6 stringclass/STRINGA_StringInfo

NAME

`STRINGA_StringInfo` -- (struct StringInfo *) ** V40 **

FUNCTION

To get a pointer to the string object's special info structure. Peeking the objects `SpecialInfo` field will not work! You must `OM_GET` this attribute.

APPLICABILITY

(G) .
