BSPrinter

Reference

Table of contents

Constants 329

PrintPreview control

This is the main object of the component.

To have print preview in a program you need to do these simple steps:

Add the necessary code to replace the Printer object to a bas module:

```
Public Property Get Printer() As Printer
    Set Printer = BSPrinter.PrinterReplacement
End Property

Public Property Set Printer(nPrinter As Printer)
    Set BSPrinter.PrinterReplacement = nPrinter
End Property
```

Then add a **PrintPreview** control to a form.
Double click on the control, and write this code:

```
Private Sub PrintPreview1_PrepareReport(Cancel As Boolean)
    MyPrintingProcedure
End Sub

Private Sub MyPrintingProcedure()
    Printer.Print "Hello Word"
    ' continue with the printing code
End Sub
```

Then add a command button (Command1) and this code:

```
Private Sub Command1_Click()
    PrintPreview1.ShowPreview
End Sub
```

Properties:

AllowUserChangeMargins *property*:

Parameters information:
Type: **BSAllowUserChangeMarginsConstants**

Description:

Returns or sets a value that determines whether the user will be able to change the margins from the Page Setup dialog box.

Possible values:
   0 - bsCMNo: The user will not be able to change the margins. The controls for changing margins in the **Page setup dialog box** will be disabled.
   1 - bsCMYes: The user will be able to change the margins. The controls for changing margins in the **Page setup dialog box** will be enabled.
   2 - bsCMWhenHandlingMargins: The user will be able to change the margins if the **HandleMargins** property is set to **True**.

The default setting is **bsCMWhenHandlingMargins**.

AllowUserChangeOrientation *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user will be able to change the page **Orientation** from the **Print Preview and Page Setup dialog boxes**. He will still be able to change it from the **Printer dialog box**. For not displaying the **Printer dialog box**, use the **AllowUserChangePrinter** property (he won't be able to choose the printer device either).

The default setting is **True**.

AllowUserChangePageNumbers *property*:

Parameters information:
Type: **BSAllowUserChangePageNumbersConstants**

Description:

Returns or sets a set of flags that determine whether the user will be able to change the page numbers settings such as **font, position, format and color**, even removing page numbers entirely.

The default setting is **bsCPAllowChangeAll**.

The flags can be used together with the OR operator, such as **bsCPAllowChangeFont** OR **bsCPAllowChangeColor**. In this case the user won't be able to change the position and format.

AllowUserChangePaperSize *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user can change the paper size from the **Page Setup dialog box**. He will still be able to change it from the **Printer dialog box**. For not displaying the **Printer dialog box**, use the **AllowUserChangePrinter** property (he won't be able to choose the printer device either).

The default setting is **True**.

AllowUserChangePrinter *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user will be able to change the Printer in the **Printer dialog box**.
If set to **False**, when the user clicks the Print button at the toolbar, the report will be sent to the printer immediately without displaying the **Printer dialog box**.

The default setting is **True**.

AllowUserChangeScale *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user can change the scale of the report. If set to **False** the controls that let the user to change the scale will not be visible in the **Print Preview dialog box**.
Changing the scale makes everything in the document smaller or bigger.

The default setting is **True**.

AuxiliaryButtonToolTipText *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the ToolTip text of the optional auxiliary button at the toolbar of the **Print preview dialog**.

AuxiliaryButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a custom button will appear on the top toolbar of the **Print Preview dialog box**.

The default setting is **False**.

This button can be used by the client program to perform an action such as displaying a custom dialog or some other task or configuration.
When it is clicked it raises the **AuxiliaryButtonClick** event.

BottomMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the bottom margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

BottomMarginEffective *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending on the setting of the **MarginUnits** property, which indicates the bottom margin actually applied. This value may be greater than the one set in the **BottomMargin** property, since the margin space can be increased if necessary to print the page numbers or footer.

BottomNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the bottom physical space where the Printer device cannot print. The **BottomMargin** cannot be set smaller than that.

Canceled *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value that indicates if the last **ShowPrinter**, **ShowPageSetup**, **ShowPreview** or **ShowSaveToFile** action was canceled or not.
With **SaveToFile**, it returns **True** if it could not save the file at the location or some other error occurred.

Notes:

PrintDirect can also produce it to be set to True when the printer driver raises an error or the **CancelPrint** argument of the **BeforePrint** event was set to **True**.

Collate *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether multiple copies of the report will be printed collated or not. Collating is performed when the number of copies is greater than 1. The Printer device must support collating, otherwise this setting has not effect.

When the copies of the report are printed collated, the pages are printed in this order: 1, 2, 3, 1, 2, 3, 1, 2, 3. And when they are not collated: 1, 1, 1, 2, 2, 2, 3, 3, 3.

The default is **True**.

This setting can also be changed by the user from the **Printer dialog box**.

ColorSupport *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value that indicates whether the printer device can print in color.

CurrentPageInPreview *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the first page visible in the **Print Preview dialog box**.

Note:

This is not necessarily the page number, but the index or position of the page, because the page number may not be the same because of the value of the property **StartingPageNumber** or because there is more than one document in the report due to a **Printer.EndDoc** in the printing procedure.

Flags *property*:

Parameters information:
Type: **BSPrinterFlagsConstants**

Description:

Returns or sets options for the **Printer dialog box**.

FontWidth *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets the font width of the printer object. The font width is generally chosen by Windows to match the selected font height and provide a standard aspect ratio, however by setting this property you can specify fonts that are narrower or wider than the default font.

FormatButtonVisible *property*:

Parameters information:
Type: **BSAutoYesNoConstants**

Description:

Returns or sets a value that determines whether a Format button will be visible at the top toolbar of the **Print Preview dialog box** to display the **ShowFormatOptions** dialog box.

The default setting is **bsAutoYN**.

bsAutoYN: The Format button will be visible if a Grid was printed using the **PrintGrid** method (* read details below).
bsNo: The Format button will not be visible.
bsYes: The Format button will be visible regardless a Grid was printed or not. From its dialog the user can also configure the report in grayscale.

Notes:

If the Format button is not visible, but **AllowUserChangePageNumbers** is set to **True**, a button to show the **ShowPageNumbersOptions** dialog will appear instead.

This Format button can also be used to display a custom **Format dialog box** by canceling the display by handling the **CancelDisplay** argument in the **BeforeShowingFormatDialog** event and showing a custom dialog instead.

Its icon and ToolTip text can be customized (see **ToolbarButtonPicture** property and **GetLocalizedText** event).

(*) If more that one grid were printed (using the **PrintGrid** method) in the report, if all the grids were printed with the same style (**GridReportStyle**), the format dialog will appear with the grid style selector visible. But if the grids were printed with different styles, the grid style selector will not be visible.

FromPage *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets a value that determines the **first page to be printed**. It will be set in the **From** text box of the **Printer dialog box**.

GDIPlusGraphics *property*:

Parameters information:
Type: **Long** (Read only)

Description:

Returns a handle to a GDI+ Graphics object that can be used to draw with that technology.

For drawing with GDI+, use this Graphics object instead of the one obtained with the **GdipCreateFromHDC** API, because if doing so, there can be problems with transparencies and device printers that use Postscript because the component uses metafiles.

Notes:
Do not make a cache copy of the handle obtained with this property because it changes for every page. As a side note, the same care must be taken with the **Printer.hDC** property of the replacement **Printer** object.

When you finished drawing, do not delete this object with the GdipDeleteGraphics API. It is deleted by the component at a proper moment when it is no longer needed.

GridReportStyles *property*:

Parameters information:
Type: **GridReportStyles** (Read only)

Description:

Collection of **GridReportStyle** that are available.

HandleMargins *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the margins will be handled automatically by the component or not (they will be handled by the client program).

The default setting is **True**.

Note:
When **HandleMargins** is **True**, the **Printer.ScaleWidth** and **Printer.ScaleHeight** properties return the space within the margins.
Also, **Printer.CurrentX** and **Printer.CurrentY** start at the end of the Left and Top margins respectively.

LeftMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the left margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

LeftNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the left physical space where the Printer device cannot print. The **LeftMargin** cannot be set smaller than that.

LetPrintExceedMargin *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the **Printer.Print** and
PrintPreview.PrintEx methods will be able to print beyond the right margin (in
RightToLeft it is the left margin).

Note: in the **PrintEx** method, if the **Alignment** parameter is set, and this property is set to
True, it has no effect.

MarginsButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to allow the user to enter into a visual mode for configuring the margins will be visible at the top toolbar.

The default setting is **False**.

Notes:

If the property setting **AllowUserChangeMargins** allows the user to change the margins, by pressing this button, the margins can be configured by dragging them.

If the property setting **AllowUserChangeMargins** does not allow the user to change the margins, pressing this button will show the margins but will not allow them to be changed.

If this button is not visible, the user will still be able to enter this visual mode for seeing/changing the margins by means of the menu that appears by right clicking over the top toolbar or by clicking the menu button at the end of the top toolbar (see **MenuButtonVisible** property), in case it is visible.

The **MarginsButtonVisible** toggles the property **ShowMargins True** and **False**.

MarginUnits *property*:

Parameters information:
Type: **BSMarginUnitsConstants**

Description:

Returns or sets a value that determines the units, **Millimeters** or **Inches**, that the program (not for the final user, for the final user see **MarginUnitsForUser**) will use for the property settings related to margins, which are: **LeftMargin**, **TopMargin**, **RightMargin**, **BottomMargin**, **MinLeftMargin**, **MinTopMargin**, **MinRightMargin**, **MinBottomMargin**, **LeftNonPrintableArea**, **TopNonPrintableArea**, **RightNonPrintableArea**, **BottomNonPrintableArea**, **TopMarginEffective** and **BottomMarginEffective**.

The default setting is **vbMillimeters**.

MarginUnitsForUser *property*:

Parameters information:
Type: **BSMarginUnitsForUserConstants**

Description:

Returns or sets a value that determines the units, **Millimeters** or **Inches**, that will be used for the margins in the **Printer dialog box**.

The default setting is **bsMUUserLocale**, that means that it will use Millimeters or Inches based in the Windows locale user setting.

MaxScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the maximum allowed value in percentage that the user can set for the Scale in the Print Preview dialog box.

MenuButtonIcon *property*:

Parameters information:
Type: **BSMenuButtonIconConstants**

Description:

Returns or sets a value that determines the icon of the menu button (at the end of the top toolbar, see **MenuButtonVisible** property).

The default setting is **bsIconStyleVerticalEllipsis**.

MenuButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to display a menu will be visible at the end of the top toolbar.

The default setting is **True**.

Notes:

Even if this property setting is to show the button, the visibility of this button will also depend on the width of the screen, whether there is enough space to display the button or not.

The menu can also be displayed, even if this button is not visible, by right clicking on a free space of the top toolbar.

MinBottomMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum bottom margin that the user can set.

The default value is **0**.

MinLeftMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum left margin that the user can set.

The default value is **0**.

MinRightMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum right margin that the user can set.

The default value is **0**.

MinScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the minimum allowed value in percentage that the user can set for the Scale in the **Print Preview dialog box**.

MinTopMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum top margin that the user can set.

The default value is **0**.

PageCount *property*:

Parameters information:
Type: **Long** (Read only)

Description:

Returns the total number of pages.

PageNumbersColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color that will be used to print the page numbers.
This font color setting can also be changed by the final user from the **Page Numbers Options or Format dialog boxes** if **AllowUserChangePageNumbers** is set to **True** (it is the default setting).

The default color is **0 - vbBlack**.

PageNumbersFont *property*:

Parameters information:
Type: **Font**

Description:

Returns or sets the Font that will be used to print the page numbers.
This font setting can also be changed by the user from the **Page Numbers Options or the Format dialog boxes** if **AllowUserChangePageNumbers** is set to **True** (it is the default setting).

The default Font is **Arial 12 Bold**.

PageNumbersFormat *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a string that determines how the page numbers will be formatted.

These characters strings have special meaning, and will be replaced with:
   {#}: page number
   {##}: total page count
   {r}: page number in roman numbers, lowercase
   {Nr}: total page count in roman numbers, lowercase
   {R}: page number in roman numbers, uppercase
   {NR}: total page count in roman numbers, uppercase
   {a}: page number using letters as numbers, lowercase
   {Na}: total page count using letters as numbers, lowercase
   {A}: page number using letters as numbers, uppercase
   {NA}: total page count using letters as numbers, uppercase
   {i}: page number in sequence of i
   {Ni}: total page count in sequence of i
   {I}: page number in sequence of I
   {NI}: total page count in sequence of I

All other characters will be printed verbatim.
But there is one exception: if the full string is "Default". In this case the default formatting will be used, that is "{#}" (just the page number).

Notes:

This setting sets the default format, but the user can change it from the **Page Numbers Options or the Format dialog boxes**, that appear pressing the Page Numbers Options or Format toolbar buttons at the **Print Preview dialog box**. The user will be able to change it as long as the **AllowUserChangePageNumbers** property setting is set to **True** (it is the default setting).

From these dialog boxes the user can select from some predefined formats that are localized, being these formats in English:
"{#}"
"-{#}-"
"{#} / {##}"
"{#}/{##}"
"{#} of {##}"
"{r}"
"{R}"
"{a}"
"{A}"
"pg. {#}"
"pg. {#} / {##}"

"pg. {#}/{##}"
"pg. {#} of {##}"
"Page {#}"
"Page {#} / {##}"
"Page {#}/{##}"
"Page {#} of {##}"
"page {#}"
"page {#} / {##}"
"page {#}/{##}"
"page {#} of {##}"

This list can be changed using the event **GetLocalizedText**, by returning the Text parameter of TextID's:

    bsPPUI_General_PageNumbersFormatStrings_Count
    bsPPUI_General_PageNumbersFormatString_Item + N

For example, when receiving in the **GetLocalizedText** event with **bsPPUI_General_PageNumbersFormatStrings_Count** as argument for **TextID**, the **Text** parameter must return the number of different formats for page numbers. The default is "21" for English **LanguageID** (different languages may have different number of page number formatting options).

After getting that number, the **GetLocalizedText** event will be raised that number of times with **TextID** argument ranging from **bsPPUI_General_PageNumbersFormatString_Item** to **bsPPUI_General_PageNumbersFormatString_Item** + 21 - 1 (or whatever number that was returned before in **bsPPUI_General_PageNumbersFormatStrings_Count**). The **Text** parameter is used to return the format string for each option in that event procedure.

In English the default predefined page numbers format strings available are "{#}" to "page {#} of {##}" listed above.

PageNumbersPosition *property*:

Parameters information:
Type: **BSPageNumbersPositionConstants**

Description:

Returns or sets a value that determines the position of the page numbers.

The default setting is **bsPosDefault**.

Notes:

To set the distance of the page numbers from the bottom of the page or from the top of the page, use **SetFooter** and **SetHeader** methods with the **DistanceFromEdge** argument to the desired value. It is not necessary to set the other parameters.

If you want not to offer some of this position options to the user, you can hide one or more of them by setting the **PageNumbersPositionFlags** property.
If you want to prevent the user from changing the page number format at all, set the **AllowUserChangePageNumbers** property without the flag **bsCPAllowChangeFormat**.

PageNumbersPositionFlags *property*:

Parameters information:
Type: **BSPageNumbersPositionFlagsConstants**

Description:

Returns or sets flags that determine the options that will be available to the user for the position of page numbers.

Note: yo can combine the flags, or make boolean operations with them.
Fort exaple:

```
PrintPreview1.PageNumbersPositionFlags = bsPFAllAvailable And
(Not bsPFAllTopAvailabl
```

will show all positions except the top ones.

PageSetupButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to configure Page Setup will be available on the top toolbar of the **Print Preview dialog box**.

The default setting is **True**.

PageView *property*:

Parameters information:
Type: **BSPageViewConstants**

Description:

Returns or sets a value that determines the page view layout that will be selected by default in the **Print Preview dialog box**.

The default setting is **bsPVFitHeight**.

Note: in case that this property is set to **bsPVNormalSize** and the property
PageViewNormalSizeButtonVisible is set to **False**, the 'View normal size' button will be
displayed anyway at the toolbar.

PageViewNormalSizeButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to display the pages in their normal size will be visible at the top toolbar.

The default setting is **False**.

In case this button is displayed, it will be place first and next to the other view buttons: **View full width**, **View full height** and **View multiple pages**.

PreviewDialogCaption *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the text that will be displayed as the caption of the **Print Preview dialog box**.
If not set, it will show the Product name of the program & " - Print preview " (localized in the available languages).

Notes:

When running in the development environment, the client program is "Visual Basic". But it will show the proper client program name once compiled.

The part of the dialog caption localized as "Print preview" in English has the following **TextID** in the **GetLocalizedText** event:

bsPPUI_PreviewDialog_Caption

PreviewPagesAccuracy *property*:

Parameters information:
Type: **BSPreviewPagesAccuracyConstants**

Description:

Returns or sets the speed and accuracy of the page rendering in the **Print Preview dialog** box.

bsPAFastDrawing: The drawing of the pages in the **Print Preview dialog** box is faster but can be some minor issues with the text scaling in long texts.

bsPAAccurateDrawing: The drawing of the pages in the **Print Preview dialog** box is slower but shows what will be printed without any scaling mismatch. The speed of displaying the pages is fast enough on new machines but can be a bit slow on older machines.

Printed *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if the last operation of **ShowPrinter**, **ShowPreview** or **PrintDirect** resulted in the report printed.

PrintPageNumbers *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines if the component will handle the printing of the page numbers automatically.

The default setting is **True**.

QuickPrintButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to print directly to the default printer or to the printer that was last used if the user settings are being remembered (see the **RememberUserPreferences** property), without showing the **Printer dialog box**, will appear in the top toolbar.

The default setting is **True**.

Note:

The visibility of this button also depends on the width space available, if the space is not enough, this button will not be visible even it this property setting is **True**.

RememberUserPreferences *property*:

Parameters information:
Type: **BSRememberUserPreferencesConstants**

Description:

Returns or sets a value that determines whether the user settings are saved and when they are saved.

These settings include things like scale, paper size, paper orientation, selected printer device, page numbers position, etc.
The default setting is: **3 - bsSPWhenPrinted**.
See **BSRememberUserPreferencesConstants** for details on each value.

Note:

In the **PrintPreview** control, the settings are saved for each control instance, but when the same control is used to print more than one report, the settings can be saved per report. For that it is necessary to set a different **ReportID** for each report. Otherwise, the settings will be saved as a whole, all reports sharing the same last settings.
In the case of using the **IPrinterEx** interface (obtaining the object from the global property **PrintPreview**) instead of a control **PrintPreview**), it is necessary to set a **ReportID** to save user preferences (even if there is only one report), otherwise they will not be saved.

ReportID *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a String that is used time to identify the current report. It is used by the component to store the user's printing preferences for each different report.

If you print more than one report with the same control, use it to identify each report. Otherwise the user's preferences will be remembered as a whole.

Note: For the user's preferences to be remembered, **RememberUserPreferences** property setting must be set to other than **bsSPNever**

ReportName *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a String that is used to identify the print job in the Windows printer spooler.
When ReportName is set, the text of the print job will be "Product name of the client program - " & ReportName. But when it is not set just the Product name of the client program will be displayed.

If this property is set, the report name will also appear as part of the title bar caption of the **Print preview dialog** box.

Notes: When running in the IDE (the development environment), the Product name of the client program will be identified as "Visual Basic". But once it runs compiled, it will be the name of your program.

For example, if the name of the program is "My Program" and ReportName is set to "Report 1", then the print job will be identified as "My Program - Report 1".

RightMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the right margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

RightNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the right physical space where the Printer device cannot print. The **RightMargin** cannot be set smaller than that.

RightToLeftControl *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines if the **RightToLeft** property of the **Printer** replacement object can be set to **True** with no control or only with locales that support right-to-left reading as the original printer does.
The default is **True** to mimic the **VB.Printer** object behavior.

ScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the scale factor on which the report will be printed.

Note:

This property is independent from **Printer.Zoom**. Both set an scale, **ScalePercent** is intended for the user, while **Printer.Zoom** can be used by the program to set one or more pages to a different scale. The final scale is defined by both multiplied.

ShowBottomToolbar *property*:

Parameters information:
Type: **BSBooleanUserCanChangeConstants**

Description:

Returns or sets a value that determines whether the **Print Preview dialog box** presents one toolbar at the top and a navigation toolbar at the bottom, or just one toolbar at the top with all the functions accessible from there. Also, if the user will be able to change that arrangement.
The navigation toolbar is used to navigate the different pages.

The default setting is **bsTrueAndUserCanChange**, showing the bottom toolbar.

ShowFirstPageNumber *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the page number, header and footer will be printed when the page number value is 1.

The default setting is **True**

ShowMargins *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether there will be visible lines marking the margins.

The default setting is **False**.

If the **AllowUserChangeMargins** property is set to allow the user to change the margins, the user can also change the margins by dragging these lines.

If the Margins button is visible (see **MarginsButtonVisible** property), the user can toggle on and off the visibility of the margins by pressing that button.
If the button is not visible, the user can still show the margin lines by selecting a menu option.

ShowSketch *property*:

Parameters information:
Type: **BSShowSketchConstants**

Description:

Returns or sets a value that determines whether the Print Preview dialog box, while preparing the report, will show a sketch on the pages or simply empty blank pages.

The default setting is **bsSKWithLargeDocument**.

The sketch will also be displayed when the report is redrawn, for example, when the user changes the paper size, orientation or scale.

SomethingPrintedOnPage *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if there is something printed on the current page.

SomethingPrintedOnReport *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if there is something printed on the report.

StartingPageNumber *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the page number to be printed on the first page, or with a negative value on which page the page numbering will start.

The default value is **1**.

Note: If the value is less than 1, then it will set on wich page the page numbers will start. For example, if this property is set to -2, the page numbers will start on the fouth page: Fist page: -2, no page number, second page: -1, no page number either, third page: 0, still no page number, fourth page: 1, the page number is printed here, the page numbering starts on this page.

ToolbarButtonPicture *property*:

Parameters information:
Type: **Picture**

Additional parameter(s):
   In   Required   **ToolbarIconIdentifier**   As
BSToolbarIconIdentifierConstants
   In   Required   **SizeIdentifier**   As
BSIconImageSizeIdentifierConstants

Description:

Returns or sets the images that will be used as the toolbar icons in the Print Preview dialog box.
They can be bitmap or icon format.
In case of bitmaps, the color for transparency is &HFF00FF or if this color is not present, the first pixel (at 0,0 position) will be used as the transparent color.

The SizeIdentifier parameter identifies the size that is being set or retrieved.
When changed, normally all 5 images need to be supplied, one at a time.

Note: why 5 sizes of icons? The toolbar buttons have the options of three sizes: small, medium and big (see **ToolbarIconsSize**).
In addition to that, if the client program is manifested as DPI aware, the component automatically handles the icon sizes for each DPI setting. It handles these two features with 5 different icon sizes: 20x20 pixels, 24x24 pixels, 30x30 pixels, 36x36 pixels and 44x44 pixels.

If the client program is not DPI aware, only the 20x20 pixels, 24x24 pixels and 30x30 pixels images are needed.
If **ToolbarIconsSize** is set to a setting where the user can't change the size of the toolbar buttons, but the program is manifested as DPI aware, only three will be used (depending on the size if the toolbar buttons, small, medium or big, they can be 20x20/24x24/30x30 for small, 24x24/30x30/36x36 for medium or 30x30/36x36/44x44 for big).
And if the program is not DPI aware and **ToolbarIconsSize** is set to a setting where the user can't change the toolbar buttons size, only one will be used. (20x20 for small icons, 24x24 for medium or 30x30 for big).

ToolbarIconsSize *property*:

Parameters information:
Type: **BSToolbarIconsSizeConstants**

Description:

Returns or sets a value that determines the size of the toolbar icons at the **Print Preview dialog box**.

The default setting is **bsPPTIconsAutoAndUserCanChange**.
If it is set to any of the settings that end in "UserCanChange", the user can change the size of the toolbar icons by selecting an option of a menu that appears right clicking on the top toolbar or by clicking the menu button, and this setting will be remembered.

ToPage *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets a value that determines the **last page to be printed**. It will be set in the **To** text box of the **Printer dialog box**.

TopMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the
MarginUnits property, that determines the top margin that will be set in new pages. When
the current page is other than the last page, it returns the margin value of the current
page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't
be allowed to change the margins from the dialogs. If you want to set default margins and
allow the user to change them, do it before showing the dialog.

TopMarginEffective *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending on the setting of the **MarginUnits** property, which indicates the top margin actually applied. This value may be greater than the one set in the **TopMargin** property, since the margin space can be increased if necessary to print the page numbers or header.

TopNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the top physical space where the Printer device cannot print. The **TopMargin** cannot be set smaller than that.

UIButtonStyle *property*:

Parameters information:
Type: **BSButtonStyleConstants**

Description:

Returns or sets a value that determines the button style of the **Print Preview, Options and Page numbers dialog boxes** buttons and **Print Preview dialog box** toolbars.

The default setting is **bsButtonStyleWindowsStandard**.

UIFontName *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the name of the font that will be used in the custom dialogs.
If not set, it will use the system font setting for Message box dialogs.

UILanguage *property*:

Parameters information:
Type: **Variant**

Description:

Returns or sets a value that determines the **language of the GUI captions** and other texts shown to the final user.
The default setting is 0 (automatic) and it means that the language will be automatically selected between the languages currently supported by the component by using Windows user locale setting. If a language is not supported by the component, the **English** captions will be used.

The client program can supply these captions in other languages not supported by the component by coding in the **GetLocalizedText** event's procedure.
This event can also be used to override (or customize) one or more captions in one or more languages, even when the language is supported by the component.

The constants that identify the language are in the **BSLanguageIDConstants** enumeration, but only some of the languages have built-in captions inside the component. This enumeration is provided in the documentation as a reference, and need to be declared in the client program to be used as named constants.

Note: it can also accept strings like "de" for German or "en-gb" for English UK, but that only works for languages that have built-in support.

UseAltScaleIcons *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the buttons to increase and decrease the scale that appear on the toolbar of the **Print Preview dialog box** will use alternative icons.
The standard icons are "+" and "-", and the alternative icons add a letter "A" to each icon, being "A+" and "A-".
The default setting is **False**.

<u>Methods</u>:

AddWatermark *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Angle**	As Long
	Default value: 0		
In	Optional	**OpacityPercent**	As Single
	Default value: 15		
In	Optional	**ZPosition**	As
BSWatermarkZPositionConstants			Default value: 1 - bsWPForeground
In	Optional	**Color**	As Long
	Default value: 0		
In	Optional	**PositionUnits**	As
BSWatermarkPositionUnitsConstants			Default value: 0 -
bsPUPercentageOfPage			
In	Optional	**Left**	As Single
	Default value: 15		
In	Optional	**Top**	As Single
	Default value: 20		
In	Optional	**Width**	As Single
	Default value: 70		
In	Optional	**Height**	As Single
	Default value: 60		
In	Optional	**Font**	As Font
	Default value: 0		
In	Optional	**ID**	As String

Description:

Adds a watermark text that will be printed on each page.
The only required parameter is the watermark text.
Any number of watermarks can be added.
The watermarks need to be added from the procedure that prints the report. With the PrintPreview control it means that they must be added when the **PrepareReport** event is executing, that is, from inside that event.

Parameters explained.

Text: the text of the watermark. It may contain more than one line by adding line feeds (vbCrLf).

Angle: the angle in that defines the direction of the watermark, between 0 and 360 degrees. The default is 0 and means horizontal. A value of 45 produces diagonal

watermark and a value of 90 a vertical watermark.

OpacityPercent: it defines the opacity of the watermark. The value must be between 0 and 100. 0 means transparent and 100 means opaque. The default is 15.

ZPosition: defines if the watermark is placed behind the printing of the document or above. The default value is **bsWPForeground** that means above.
Note: When the watermark is placed on the foreground, it uses GDI+ that is available from Windows XP.

Color: sets the fore color of the text. The default value is **vbBlack**

PositionUnits: defines the units used in the following position and size parameters.
    bsPUPercentageOfPage: The watermark position and size are set in percentage of the page (of the full page, without taking into account margins, but taking into account the non-printable area).
    bsPUPrinterScaleMode: The watermark position and size are set in the current ScaleMode units of the Printer object. If HandleMargins is True, the position and size is set inside the margins. Printer.ScaleWidth and Printer.ScaleHeight return the full ranges as a reference.

Left, **Top**, **Width** and **Height**: define the position of the watermark in the page. The default values are: 15, 20, 70 and 60, that along with **PositionUnits** set to **bsPUPercentageOfPage** (also the default value) position the watermark in the middle of the page.

Font: sets the font. The size is automatically changed to fit the space that was defined with the position parameters.

ID: needed only if you are going to call the **RemoveWatermark** method. It is used to identify the watermark. More than one watermark can share the same **ID** (in that case they will be removed together by **RemoveWatermark**.

Note: for printing the watermark in right to left direction, the **Printer** must be set with **RightToLeft** to **True** before adding the watermark.

CapturePrinterObject *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

It will tell that you will start using the Printer object for this control. It can be needed to do in order to set some initial condition outside the **PrepareReport** event as the default Orientation, PaperSize, ColorMode, Etc.

ClearPage *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Clears anything that has been printed on the current page without removing the page.

The current page can be known from **Printer.Page**.

DeleteAllUserPreferences *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Deletes all the reports user preferences that were previously saved for all the application.

Note:

The custom styles (**GridReportStyle**) that the user saved are not deleted.

You can delete them with this code:

```
Dim Style As GridReportStyle

For Each Style In PrintPreview1.GridReportStyles
    If Style.Custom Then
        PrintPreview1.GridReportStyles.Remove Style
    End If
Next
```

DeleteUserPreferences *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Optional   **ReportID**   As String

Description:

Deletes the user preferences that were previously saved.

For the **PrintPreview** control, if no **ReportID** is set (in this property), the last settings are saved per control instance. If **ReportID** property setting is set, they are saved per control instance + **ReportID**.

The optional parameter **ReportID** can be used to delete the settings of a different report than the one currently set in the **ReportID** property (if any).

Note:

The custom styles (**GridReportStyle**) that the user saved are not deleted.

GetCurrentGridReportStyle *method*:

Parameters information:
Return Type: **GridReportStyle**
No parameters.

Description:

Returns the **GridReportStyle** that will be used to draw the grid(s) on the report.
In the printing procedure it can be used to draw things with the style (colors and other properties) chosen by the user.

GetDocFirstPageIndex *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the page index where a document starts.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page index of the first page of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the index of the first page of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetDocLastPageIndex *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the page index where a document ends.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page index of the last page of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the index of the last page of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetDocPageCount *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
    In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the number of pages of a document.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers and page count, and by calling this method it is possible to get the page count of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the page count of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetFooterPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
   In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the footer in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not have a footer, it returns -1.

GetHeaderPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
   In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the header in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not have a header, it returns -1.

GetLastDeviceName *method*:

Parameters information:
Return Type: **String**
No parameters.

Description:

Returns the last device name that is saved as user preference for the current **PrintPreview** control or **ReportID** setting.

GetPageNumber *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **PageIndex**   As Variant

Description:

Returns the page number for a given page index. If **PageIndex** is omitted, it returns the page number of the current page.

Notes:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page number of any page. The parameter **PageIndex** must have a number between **1** and **PageCount**.

The first page number can be set to other than 1 using the **StartingPageNumber** property. If there is more than one document in the report, the **StartingPageNumber** setting will be applied only to the first one.

GetPageNumberPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
    In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the page number in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not print the page number, it returns -1.

GoToPage *method*:

Parameters information:
Return Type: None.
Parameter(s):
    In   Required   **PageIndex**   As Long

Description:

By calling this method, the current page can be set to any of the already printed pages.

Notes:

If the **PageIndex** argument is set to a non-existent page, an error 9 (subscript out of range) will happen.

If, when printing text on a page, the end of a page is reached, when printing more text, it will be added to the next page.
You can use any of the graphics methods on the page, and if a new page is added by calling **Printer.NewPage**, the new page will be inserted following the current page.

In the case that the object holds more than one document (that can happen if an **Printer.EndDoc** statement was added in the middle of the printing routine), the **PageIndex** argument may not correspond to the page number. That also can happen if the **StartingPageNumber** property is set to other than 1.

PrintDirect *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Sends the report directly to the printer, without displaying the **Print Preview or Printer dialog boxes**.

Note:

If **RememberUserPreferences** is set to other than **bsSPNever**, and the user settings have been saved, it will be printed with the last user selected settings.

PrinterExists *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):
    In   Required   **DeviceName**   As String

Description:

It returns **True** if the printer pointed by **DeviceName** exists, but does not select the printer.

PrintEx *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**Text**	As Variant
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Width**	As Variant
In	Optional	**Height**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
In	Optional	**CancelIfTextExceedsWidth**	As Boolean
	Default value: False		
Out	Optional	**TextExceedsWidth**	As Boolean
Out	Optional	**StartsNewPage**	As Boolean

Description:

Like the **Printer.Print** method, it allows to print text but with some additional options.

Parameters explained:

Text: the text to be printed

Options: can be any of these values:
   bsLineFeed = 0
   bsNoLineFeed = 1
   bsAddTab = 2
The default for this optional parameter is **bsLineFeed**, that means that like the original **Printer.Print** method, it will add a line feed automatically after the text.
On the contrary, if this parameter is set to **bsNoLineFeed**, then no line feed will be added. It is like the function of the semicolon (;) in the original **Printer.Print** method.
bsAddTab adds a Tab after the text, just like the comma (,) in the Printer.Print method.

X, **Y**, **Width** and **Height**: These parameters can define the rectangle on the page where the text will be printed. If they are omitted, the **Printer.CurrentX** and **Printer.CurrentY** will be used for the position, and the rectangle is the available space on the page. The text

can wrap to fit in the rectangle.

Alignment: Sets the alignment of the text in the available space. If the rectangle was set by the **X**, **Y**, **Width** and **Height** arguments, the text will be aligned in that rectangle according to the setting of this parameter.
Possible values are:
   bsAlignLeftTop = 0
   bsAlignLeftCenter = 1
   bsAlignLeftBottom = 2
   bsAlignCenterTop = 3
   bsAlignCenterCenter = 4
   bsAlignCenterBottom = 5
   bsAlignRightTop = 6
   bsAlignRightCenter = 7
   bsAlignRightBottom = 8
The default setting for this parameter is **bsAlignLeftTop**

WordWrap: determines whether the text will wrap into another line if it is too long to fit the available horizontal space.
The default setting for this parameter is True.
Note: This is one of the main advantages over the original **Printer.Print** method.

WordBreak: similar to WordWrap, but in this case it determines whether the words can be split into more than one line when the text doesn't fit into the available horizontal space.
The default setting for this parameter is **False**.
Note: the difference between **WordWrap** and **WordBreak**, is that **WordWrap** does not split words. **WordBreak** implies **WordWrap**.

RightToLeftReading: Sets the text direction right-to-left or left-to-right without changing the RightToLeft property setting.
If set to **bsAutoYN** (default setting), the RightToLeft property setting is used instead to set the text direction.
Note: **True** is converted to **bsYes**

RectMarginsX, **RectMarginsY**:
These values can define margins within the rectangle set by the parameters **X**, **Y**, **Width** and **Height**; therefore, they affect the position of the text that is according to the Alignment parameter.
They must be in the current ScaleMode of the Printer.

CancelIfTextExceedsWidth:
Determines whether the printing operation will be aborted if the horizontal space is smaller than required.
The default setting for this parameter is **False**.

TextExceedsWidth: This Boolean parameter needs to be passed **by reference** and **is a return value**. Returns **True** if the required horizontal space is greater than the available space.

StartsNewPage: This Boolean parameter needs to be passed **by reference** and **is a**

return value. Returns **True** if this print operation starts a new page.

PrintForm *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**Form**	As Object
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**OnlyClientArea**	As Boolean
	Default value: False		
In	Optional	**ScaleFactor**	As Single
	Default value: 1		
In	Optional	**CenteredHoriz**	As Boolean
	Default value: False		
In	Optional	**ShrinkToFit**	As Boolean
	Default value: True		
Out	Optional	**PrintedX**	As Variant
Out	Optional	**PrintedY**	As Variant
Out	Optional	**PrintedWidth**	As Variant
Out	Optional	**PrintedHeight**	As Variant

Description:

Prints the image of a form.

It is similar to **PrintWindow**, but the first argument accepts a **Form** object instead of a window handle.

Parameters:

Form: form to be printed.

X: the X position where the image will be printed. If not supplied, CurrentX is used.

Y: the Y position where the image will be printed. If not supplied, CurrentY is used.

OnlyClientArea: If **True**, only the client area will be printed (not the title bar and borders)

ScaleFactor: Scale of the printed image, the default is 1.

PrintedX Returns the X position where the image was printed.

PrintedY Returns the Y position where the image was printed.

PrintedWidth Returns the Width of the printed image.

PrintedWidth Returns the Height of the printed image.

Notes:

It requires Windows XP or later.
Controls that use GDI+ may not appear.

Printer.CurrentX won't be updated, but **Printer.CurrentY** will be positioned after the image.

PrintGrid *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):

In	Required	**Grid**	As Object
In	Optional	**Style**	As Variant
In	Optional	**UserCanChangeStyle**	As Boolean

     Default value: True

| In | Optional | **GridPosition** | As **BSGridPositioningConstants** |

     Default value: 0 - bsGACentered

| In | Optional | **ColsToHide** | As Variant |
| In | Optional | **PrintCellsFormatting** | As |

BSPrintCellsFormattingConstants   Default value: 0 - bsPCFPrintAll

| In | Optional | **IgnoreEmptyRowsAtTheEnd** | As Boolean |

     Default value: True

Description:

Prints the contents a grid.
Supported types are **MSFlexGrid**, **MSHFlexGrid** and compatibles.

Parameters explained:

Grid: A reference to the Grid control to be printed. This is the only parameter that is required.

Style: This optional parameter can set the Style of the Grid. It can be an object **GridReportStyle**, a number identifying a Style or a String with the Name of a Style.
The style can be one built-in, or one that was added by calling the **Add** method of the **GridReportStyles** collection.

UserCanChangeStyle: Determines if the user can customize the look of the printed grid by selecting another Style.
Its default value is **True**.

GridPosition: Determines the position of the grid if it is smaller than the page width.
Possible values are:
   bsGACentered = 0
   bsGALeft = 1
   bsGARight = 2
   bsGAStretch = 3
   bsGAEnlargeLastColumn = 4
   bsGACenteredAndUserCanChange = 8
   bsGALeftAndUserCanChange = 9
   bsGARightAndUserCanChange = 10
   bsGAStretchAndUserCanChange = 11
   bsGAEnlargeLastColumnAndUserCanChange = 12
The default value is **bsGACentered**.

ColsToHide: optional array of Boolean values that determines what columns to hide (not to print).

PrintCellsFormatting: Determines whether the cells formatting will be printed and what. Possible values are:
    bsPCFPrintAll = 0
    bsPCFIgnoreForeColor = 1
    bsPCFIgnoreBackColor = 2
    bsPCFIgnoreFonts = 4
    bsPCFIgnoreTextAlignment = 8
    bsPCFIgnorePictures = 16
    bsPCFIgnoreAll = 31
The default value is **bsPCFPrintAll**.

IgnoreEmptyRowsAtTheEnd: If there are empty rows at the end of the grid (not filled with data), it determines whether these rows will printed or not.
The default value is **True** (that means that the empty rows at the end will not be printed).

Return value:
When the grid has many rows, the **Print Preview dialog box** will show a window with the progress of loading the data and preparing the preview, and it will show a button that allows to cancel the operation. If the user clicked the button to cancel the operation, the **Print Preview dialog** won't be displayed and this method will return **False**, meaning that the operation was canceled. In a normal operation it returns **True**.

PrintRichTextBox *method*:

Parameters information:
Return Type: None.
Parameter(s):
    In   Required   **RTBControl**   As Object

Description:

Prints the contents of a RichTextBox control.

The parameter **RTBControl** is a reference to the Rich TextBox control to be printed.

PrintWindow *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**hWnd**	As Long
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**OnlyClientArea**	As Boolean
	Default value: False		
In	Optional	**ScaleFactor**	As Single
	Default value: 1		
In	Optional	**CenteredHoriz**	As Boolean
	Default value: False		
In	Optional	**ShrinkToFit**	As Boolean
	Default value: True		
Out	Optional	**PrintedX**	As Variant
Out	Optional	**PrintedY**	As Variant
Out	Optional	**PrintedWidth**	As Variant
Out	Optional	**PrintedHeight**	As Variant

Description:

Prints the image of a window given its handle.
The window can be a control, a form, or any window that has a DC.
The printed image is how the control or window appears on the screen.
Windowless controls, such as Label or Shape, cannot be printed using this method.
The control must be visible, but it can be placed outside the visible area or covered with other controls.

Parameters:

hWnd: handle of the window to be printed.

X: the X position where the image will be printed. If not supplied, CurrentX is used.

Y: the Y position where the image will be printed. If not supplied, CurrentY is used.

OnlyClientArea: If **True**, only the client area will be printed (not the title bar and in some windows types the borders and scroll bars)

ScaleFactor: Scale of the printed image, the default is 1.

PrintedX Returns the X position where the image was printed.

PrintedY Returns the Y position where the image was printed.

PrintedWidth Returns the Width of the printed image.

PrintedWidth Returns the Height of the printed image.

Notes:

It requires Windows XP or later.

Printer.CurrentX won't be updated, but **Printer.CurrentY** will be positioned after the image.

ReleasePrinterObject *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Releases the Printer object from the current control.

Note: if you used **CapturePrinterObject** for one control and now need to set it to another control, it is not necessary to do a release first (it is done automatically).

RemoveAllWatermarks *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Removes all the watermarks.

Notes:
If you are calling **RemoveAllWatermarks** from within the printing procedure (while in the **PrepareReport** event of the **PrintPreview** control) to remove the watermarks from certain page and ahead, call it before the **Printer.NewPage** that adds that page.
Also, if you are calling **RemoveAllWatermarks** from within the printing procedure, add the watermarks by calling **AddWatermark** method inside the printing procedure too.

RemovePage *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Removes de current page.

To remove pages that are not the last one, use **GoToPage** method.
The current page can be known from **Printer.Page**.

RemoveWatermark *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Required   **ID**   As String

Description:

Removes a watermark.

The **ID** argument identifies the watermark or watermarks to remove.
If there is no watermark with this ID, it does not generate an error.

SaveToFile *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**DeviceNameToUse**	As String
In	Required	**FileFullPath**	As String

Description:

Saves the document to a file instead of printing it.
It should be used only with printers drivers that produce a meaningful output file.
The file format and its corresponding extension depends on the printer driver. Most drivers will produce an intermediate file that is used by the spooling system, such as PostScript files, but some printers driver can produce a useful output such as PDF files or XPS files.
In Windows 10, the "Microsoft Print to PDF" driver produces a PDF file, whereas "Microsoft XPS Document Writer" outputs XPS files.

Parameters explained:

DeviceNameToUse is the printer driver to be used.

FileFullPath is the full path of the output file.

Notes: Unlike the **ShowSaveToFile** method, this method saves the file without displaying any dialog unless the driver itself shows dialogs on the operation.
If the printer driver is not installed the end user will get a message telling that the file cannot be saved because the printer driver is not installed.
You can use the **PrinterExists** method to check whether a particular printer driver is installed.

SelectPrinter *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):

In	Required	**DeviceName**	As String
In	Optional	**PreserveUserPrintingPreferences**	As Boolean

      Default value: False

Description:

It selects a printer.

Parameters explained:

DeviceName: device to select.

PreserveUserPrintingPreferences: if **True**, the current printing settings such as Orientation, PaperSize, ColorMode will be preserved is the driver supports them. If the paper size is not supported by the printer, it will select the most similar paper. If **False** the settings will be printer default settings. The default value is **False**

It returns **True** if the printer was successfully selected, **False** otherwise.
The reason why a printer selection can fail is because the printer doesn't exists or because **DeviceName** was misspelled.

Note: it is similar to the code:

```
Dim prn As Printer

For Each prn In Printers
    If prn.DeviceName = DeviceName Then
        Set Printer = prn
        Exit For
    End If
Next
```

SetFooter *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**TextOrPicture**	As Variant
In	Optional	**Position**	As

BSHorizontalPositionConstants   Default value: 0 - bsHPosLeft

In	Optional	**DistanceFromEdge**	As Single

Default value: 15

In	Optional	**MultilineTextAlignment**	As

BSMultilineTextAlignmentConstants   Default value: -1 -
bsMTAlignAccordingToPosition

In	Optional	**Font**	As Variant
In	Optional	**TextColor**	As Long

Default value: 0

In	Optional	**PictureScale**	As Single

Default value: 1

Description:

Sets an image or text that will be printed as a footer on each page, such as a logo or a note.
It can be an image or a text, but not both. This setting is for all the report, it cannot be set for individual pages.

Note:

This feature is not available in the demo version.

Parameters explained:

TextOrPicture: this Variant parameter can accept a stdPicture with an image or a text. To remove a footer previously set, omit this parameter, or provide an empty Variant, an Object variable containing Nothing or a null String.

Position: sets the position of the image or text (single line text) to the left, right or horizontally centered.
    bsHPosLeft = 0
    bsHPosRight = 1
    bsHPosCenter = 2
The default value is **bsHPosLeft**.

DistanceFromEdge: sets the distance from the edge of the page to the footer.
Note that the footer and header are usually within the margin space.
The default value is **15 millimeters (0.59 inch)**.

MultilineTextAlignment: sets the alignment of the different lines of text when the text does not fit in one line or has vbCrLf in it.

Font : sets the font for the text.
Default font is **Arial 12 Regular**.

TextColor: sets the fore color of the text.
The default color is **vbBlack**.

PictureScale: sets the scale of the picture.
The default scale is **1**.

SetHeader *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**TextOrPicture**	As Variant
In	Optional	**Position**	As

BSHorizontalPositionConstants Default value: 0 - bsHPosLeft

| In | Optional | **DistanceFromEdge** | As Single |

Default value: 15

| In | Optional | **MultilineTextAlignment** | As |

BSMultilineTextAlignmentConstants Default value: -1 -
bsMTAlignAccordingToPosition

| In | Optional | **Font** | As Variant |
| In | Optional | **TextColor** | As Long |

Default value: 0

| In | Optional | **PictureScale** | As Single |

Default value: 1

Description:

Sets an image or text that will be printed as a header on each page, such as a logo or a note.
It can be an image or a text, but not both. This setting is for all the report, it cannot be set for individual pages.

Parameters explained:

TextOrPicture: this Variant parameter can accept a stdPicture with an image or a text.
To remove a header previously set, omit this parameter, or provide an empty Variant, an Object variable containing Nothing or a null String.

Position: sets the position of the image or text (single line text) to the left, right or horizontally centered.
    bsHPosLeft = 0
    bsHPosRight = 1
    bsHPosCenter = 2
The default value is **bsHPosLeft**.

DistanceFromEdge: sets the distance from the edge of the page to the footer.
Note that the footer and header are usually within the margin space.
The default value is **15 millimeters (0.59 inch)**.

MultilineTextAlignment: sets the alignment of the different lines of text when the text does not fit in one line or has vbCrLf in it.

Font : sets the font for the text.
Default font is **Arial 12 Regular**.

TextColor: sets the fore color of the text.

The default color is **vbBlack**.

PictureScale: sets the scale of the picture.
The default scale is **1**.

ShowFormatOptions *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Format dialog box**, where the user can select page numbers options, color or grayscale for the report, and if there is printed a grid, the layout and color of it.

The dialog **ShowPageNumbersOptions** is similar, but it just handles the page numbers options.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPageNumbersOptions *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Page Numbers dialog box**, where the user can select the page numbers options.

The dialog **ShowFormatOptions** is similar, but it handles a few more options, including grid layouts if there is a grid printed in the report.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPageSetup *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Page Setup dialog box**, where the user can select the paper size, page orientation and margins.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPreview *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Print Preview dialog box**.

If the user didn't print the report, the return value will be **bsDialogResultCancel**.

It returns **bsDialogResultOK** if the user printed the report or **bsDialogResultCancel** otherwise.

ShowPrinter *method*:

Parameters information:
Return Type: **BSShowPrinterDialogResultConstants**
No parameters.

Description:

Displays the **Printer dialog box**, where the user can select the printer and other options.

If the user clicked the Print button, it returns **bsPDResultPrint**, if cancelled it returns **bsPDResultCancel**, and if the user clicked the Apply button **bsPDResultApply**.

ShowSaveToFile *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
Parameter(s):

In	Required	**DeviceNameToUse**	As String
In	Required	**DefaultFileNameOrPath**	As String
In	Optional	**RememberLastPath**	As Boolean
	Default value: True		
In	Optional	**OverwritePrompt**	As Boolean
	Default value: True		

Description:

It saves the document to a file instead of printing it.
It is similar to the **SaveToFile** method, but in this case the user is asked where to save the file.

It should be used only with printers drivers that produce a meaningful output file.
The file format and its corresponding extension depends on the printer driver. Most drivers will produce an intermediate file that is used by the spooling system, such as PostScript files, but some printers driver can produce a useful output such as PDF files or XPS files.
In Windows 10, the "Microsoft Print to PDF" driver produces a PDF file, whereas "Microsoft XPS Document Writer" outputs XPS files.

Parameters explained:

DeviceNameToUse is the printer driver to be used.

FileFullPath is the full path of the output file.

Notes: Some printer drivers might show additional dialogs on the operation.
If the printer driver is not installed the end user will get a message telling that the file cannot be saved because the printer driver is not installed.
You can use the **PrinterExists** method to check whether a particular printer driver is installed.

TextHeightEx *method*:

Parameters information:
Return Type: **Single**
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Width**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
Out	Optional	**StartsNewPage**	As Boolean

Description:

Similar to Printer.TextHeight but with additional parameters.
Usually used in combination with **PrintEx** method. Please refer to **PrintEx** to read the explanation of the parameters.

TextWidthEx *method*:

Parameters information:
Return Type: **Single**
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Height**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
Out	Optional	**StartsNewPage**	As Boolean

Description:

Similar to Printer.TextWidth but with additional parameters.
Usually used in combination with **PrintEx** method. Please refer to **PrintEx** to read the explanation of the parameters.

Events:

AfterShowingPageSetupDialog *event*:

Parameters information:
No parameters.

Description:

Occurs after the **Page Setup dialog box** is displayed. Check the **Canceled** property to see whether the user changed something or not.

AfterShowingPrinterDialog *event*:

Parameters information:
No parameters.

Description:

Occurs after the **Printer dialog box** is displayed. Check the **Canceled** property to see whether the user changed something or not, and the **Printed** property to see whether the report was printed.

AuxiliaryButtonClick *event*:

Parameters information:
Parameter(s):
   Returns value   **UpdateReport**   As Boolean

Description:

Occurs when the user clicks on the Auxiliary button.

It can be used to perform certain action such as displaying a custom dialog or some other task or configuration.

UpdateReport is a return value.
The default value for the UpdateReport is True.
If it is returned with **False the report will not be redrawn, otherwise the PrepareReport** event will be raised and the pages at the **Print preview dialog** will be updated.

BeforePrint *event*:

Parameters information:
Parameter(s):
   Returns value   **CancelPrint**   As Boolean

Description:

Occurs when a report is about to be sent to the printer.
The print action can be canceled by setting the **CancelPrint** argument to **True**.

BeforePrintingPageNumbers *event*:

Parameters information:
Parameter(s):
    Returns value   **Cancel**   As Boolean

Description:

Occurs when the page numbers are about to be printed.
The page numbers printing can be canceled at this time by setting the **Cancel** argument to **True**.

BeforeShowingFormatDialog *event*:

Parameters information:
Parameter(s):
    Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Format Options dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed and you can show your own dialog instead.

BeforeShowingPageSetupDialog *event*:

Parameters information:
Parameter(s):
   Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Page Setup dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed.

BeforeShowingPrinterDialog *event:*

Parameters information:
Parameter(s):
    Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Printer dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed.

CurrentPrinterDefaultSettingChange *event*:

Parameters information:
Parameter(s):
Info	**PropertyName**	As String
Info	**NewValue**	As Variant
Info	**ControlInUse**	As Boolean

Description:

Occurs when a default setting is changed in Windows for the currently selected printer device (such as default Orientation or Paper Size).

For example if the Orientation changes from vertical to horizontal, the **PropertyName** argument will be "Orientation" and **NewValue** will be 2 - vbPRDPHorizontal

The parameter **ControlInUse** indicates whether this control instance is currently in use (with a report).

DefaultPrinterChange *event*:

Parameters information:
Parameter(s):
    Info   **ControlInUse**   As Boolean

Description:

Occurs when the default printer device is changed in Windows.

The parameter **ControlInUse** indicates whether this control instance is currently in use (with a report).

EndDoc *event*:

Parameters information:
Parameter(s):
Info	**FirstPageIndex**	As Long
Info	**LastPageIndex**	As Long

Description:

Occurs when the report was prepared but before the page numbers are printed.

GetLocalizedText *event*:

Parameters information:
Parameter(s):

Info	**LanguageID**	As Long
Info	**SubLanguageID**	As Long
Info	**TextID**	As Long
Returns value	**Text**	As String
Info	**Number**	As Long

Description:

Occurs for every caption or text that the component needs to display.

From the procedure of this event it is possible to translate the texts into any language that does not have built-in support by the component.

The **LanguageID** argument identifies the language. If the **UILanguage** property is set to **0 - bsLang_AUTO_SELECT** (default setting), the **LanguageID** argument will have the code number of the actual user locale, but if that property is set to a specific locale code number, it will come always with that number. The list of possible values are in the **BSLanguageIDConstants** enumeration.

SubLanguageID: some languages can have a sub-language ID, it will be according to the locale of the user.
SUBLANG_ENGLISH_UK = 2
SUBLANG_PORTUGUESE_BRAZILIAN = 1
(the above two are already supported without doing anything)
SubLanguageID is intended to be used together with the language **LanguageID**.
You can ignore the **SubLanguageID** parameter if you want to provide the same **Text** for all locations of the language.

The **TextID** argument identifies what caption is requested. The list of possible values with the corresponding texts in English are in the **BSUserInterfaceTextsIDConstants** enumeration.

The way to customize a text is to assign it to the **Text** parameter. It comes with a default text but it is a return parameter.

Number: This parameter is reserved, currently not used.

Notes:

This event can be used to customize the texts, even if the language has built-in support for that language by the component.

For languages that do not have built-in support, English captions will be used.

When translating to a new language, many of them are standard texts that can be found with **https://www.microsoft.com/en-us/language/search**

OrientationChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the user changes the page Orientation.

PaperSizeChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the user changes the paper size.

PrepareReport *event:*

Parameters information:
Parameter(s):
    Returns value   **Cancel**   As Boolean

Description:

Event to call the code to print the report using the **Printer** object replacement. The report must be printed on the **Printer** object replacement from within this event procedure. This is the main event.

ReportPrinted *event*:

Parameters information:
No parameters.

Description:

Occurs after a report is sent to a printer device.

ScaleChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the scale is changed by the user from the **Print Preview dialog box**.

StartDoc *event*:

Parameters information:
Parameter(s):
    Info   **FirstPageIndex**   As Long

Description:

Occurs when in a document is added the first page.

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, the **FirstPageIndex** argument has the page index of the first page of the current document.

StartPage *event*:

Parameters information:
No parameters.

Description:

Occurs every time a new page is added to the report.

Global object

Global class.

This class is automatically instantiated and its properties can be accessed without qualification.

<u>Properties</u>:

PrinterReplacement *property*:

Parameters information:
Type: **Object**

Description:

Returns or sets the new **Printer** object. It replaces the original **VB.Printer** object.

It is required in the client program (your program) that this code be added in a bas module:

```
Public Property Get Printer() As Printer
    Set Printer = PrinterReplacement
End Property

Public Property Set Printer(nPrinter As Printer)
    Set PrinterReplacement = nPrinter
End Property
```

If the program only has one form, the code can be added just on the form and the procedures can be **Private** instead of **Public**.

PrintPreview *property*:

Parameters information:
Type: **IPrinterEx** (Read only)

Description:

Returns an object similar to a **PrintPreview** control, with the same properties and methods, but can be used in Class modules as it does not require a control container to sit on. It can also be used in forms, but a **PrintPreview** control is simpler to use on them.

Notes:

When the component is registered, before using this object you'll need anyway to load a form with a **PrintPreview** control at the start of the program to load the license information, otherwise it will work unregistered. The form not need to be shown and can be unload after that.

To call the printing procedure, you can do it in two ways, by **using events** or by setting the printing procedure.
Please refer to the **IPrinterEx** object for an explanation of the options. Also the **PrinterExEvents** object has information of the first option and the **SetPrintingProcedure** of the second.

GridReportStyle object

Object that defines a style for printing a Grid control.

Properties:

ColumnsDataLinesColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color of the columns lines (vertical) in the data section (not the headers), in case that they are visible according to **DrawColumnsDataLines** property setting.

ColumnsHeadersLinesColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color of the columns lines (vertical) in the headers section (not the data), in case that they are visible according to **DrawColumnsHeadersLines** property setting.

Custom *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns **True** if the Style was created by the user through customization in the **Format dialog box**.

DrawColumnsDataLines *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the lines of the columns in the data section will be drawn or not.

DrawColumnsHeadersLines *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the lines of the columns in the header section will be drawn or not.

DrawFixedColsBackground *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or set a value that determines whether the background of the fixed columns will be painted or not. When this setting is **True**, it will be painted with the same color of the headers, set in the **HeaderBackgroundColor** property.

The default setting for all the built-in styles is **False**.

DrawHeaderBackground *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the headers background will be painted or not. When this setting is **True**, it will be painted with the with the color set in the **HeaderBackgroundColor** property.

DrawHeaderBorder *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the outer border of the headers section will be drawn or not. In case it is **True**, it will be drawn with the color set in the **HeaderBorderColor** property.

Note:

The thickness of the lines is set by the **LineThickness** property.

DrawHeaderSeparatorLine *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether it will draw a separator line between the headers section and the data section. This line will be draw with the color set in **HeaderBorderColor** property, and the thickness of the line is set by the **HeaderSeparatorLineThickness** property.

Note:

If **DrawHeaderBorder** is set to **True, this line can overlap the headers border.**

DrawOtherBackgrounds *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or set a value that determines whether the background colors of the data cells will be painted or not. In case it is **True**, they will be painted with the original colors as they are in the grid control.

DrawOuterBorder *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the outer border of the whole grid will be drawn or not. In case it is **True**, it will be drawn with the color set in **OuterBorderColor** property.

DrawRowLines *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the horizontal lines that defines the rows will be drawn or not. In case it is **True**, it will be printed with the color set in the **RowLinesColor** property.

HeaderBackgroundColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the background color of the headers section, in case it is visible according to **DrawHeaderBackground** property setting.

HeaderBorderColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color of the outer border of the whole headers section, in case it is visible according to **DrawHeaderBorder** property setting.

HeaderSeparatorLineThickness *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the thickness of the line that separates the header section from the data section, in case that it is visible according to **DrawHeaderSeparatorLine** property setting.

LineThickness *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the thickness of the lines that draws the cells.

Note:

The lines can be visible or not according to the settings of these properties:
DrawColumnsHeadersLines, **DrawColumnsDataLines** and **DrawRowLines**.

Name *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the name of the Style.

Note:

The name is localized (translated into different languages) and can be customized through the **GetLocalizedText** event.

See their **TextID**'s in **BSUserInterfaceTextsIDConstants**, they start with bsPPUI_General_GridReportStyleNames_*

OuterBorderColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color of the outer border of the grid, including headers and data sections, in case it is visible according to **DrawOuterBorder** property setting.

RowLinesColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color of the rows lines (horizontal) in the data section (not the headers), in case that they are visible according to **DrawRowLines** property setting.

Methods:

Clone *method*:

Parameters information:
Return Type: **GridReportStyle**
No parameters.

Description:

Returns a new GridReportStyle object that is a copy of this one, with the same property values.

GridReportStyles object

Collection of the **GridReportStyle** objects that are available.

The built-in collection includes there styles:
(the names are listed in English but they are localized in the languages that are available)

Basic
Basic Gray
Basic Red
Basic Green
Basic Yellow
Modern
Modern Gray
Modern Red
Modern Green
Modern Yellow
Grid
Grid Gray
Grid Red
Grid Green
Grid Yellow
Closed
Closed Gray
Closed Red
Closed Green
Closed Yellow
Horizontal
Horizontal Gray
Horizontal Red
Horizontal Green
Horizontal Yellow
Free
Free Gray
Free Red
Free Green
Free Yellow

New styles can be added with the **Add** method.
The user can also define new styles by customizing them in the **Format dialog box**.

[Properties]:

Count *property*:

Parameters information:
Type: **Long** (Read only)

Description:

Returns the number of styles the collection has.

Methods:

Add *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**NewStyle**	As **GridReportStyle**
In	Optional	**Before**	As Variant
In	Optional	**After**	As Variant

Description:

Add a new style to the collection. The style object must be previously created.

Example code:

```
Dim Style As GridReportStyle

Set Style = New GridReportStyle

Style.DrawColumnsDataLines = False
Style.DrawRowLines = True
Style.RowLinesColor = vbCyan
Style.Name = "My style"
PrintPreview1.GridReportStyles.Add Style
```

The **CreateNew** method also adds a new style, but creates it in one line by setting its properties.

CreateNew *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**Name**	As String
In	Optional	**DrawOuterBorder**	As Boolean

Default value: False

In	Optional	**DrawHeaderBorder**	As Boolean

Default value: True

In	Optional	**DrawColumnsDataLines**	As Boolean

Default value: True

In	Optional	**DrawColumnsHeadersLines**	As Boolean

Default value: True

In	Optional	**DrawRowLines**	As Boolean

Default value: False

In	Optional	**DrawHeaderSeparatorLine**	As Boolean

Default value: True

In	Optional	**DrawHeaderBackground**	As Boolean

Default value: True

In	Optional	**DrawFixedColsBackground**	As Boolean

Default value: False

In	Optional	**DrawOtherBackgrounds**	As Boolean

Default value: True

In	Optional	**LineThickness**	As Long

Default value: 3

In	Optional	**HeaderSeparatorLineThickness**	As Long

Default value: 3

In	Optional	**OuterBorderColor**	As Long

Default value: 13567272

In	Optional	**ColumnsDataLinesColor**	As Long

Default value: 16622257

In	Optional	**ColumnsHeadersLinesColor**	As Long

Default value: 16622257

In	Optional	**RowLinesColor**	As Long

Default value: 16703718

In	Optional	**HeaderBorderColor**	As Long

Default value: 13567272

In	Optional	**HeaderBackgroundColor**	As Long

Default value: 16775668

In	Optional	**Before**	As Variant
In	Optional	**After**	As Variant

Description:

Creates a new style with the properties stated in the arguments.
It is similar to the **Add** method, but instead of working with a **GridReportStyle** object, it works with parameters.

Example code:

```
PrintPreview1.GridReportStyles.CreateNew "My style", , , False, ,
True, , , , , , , , , , , vbCyan
```

Remove *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Required   **IndexNameOrObject**   As Variant

Description:

Removes a style from the collection.

The **IndexNameOrObject** argument must provide **a String** with the name of the style to remove or **a number** with its index (position) in the collection.

StylesAreEqual *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):
   In   Required   **GridReportStyle1**   As **GridReportStyle**
   In   Required   **GridReportStyle2**   As **GridReportStyle**

Description:

Returns **True** if the two styles supplied by the **GridReportStyle1** and **GridReportStyle2** parameters are equal.

IPrinterEx object

Global object that can be used instead of a PrintPreview control, for example in class modules.

This object uses this interface, which is returned by the class **Global** from its property **PrintPreview**.
It has the same properties and methods of the **PrintPreview** control, but it is a global object and it doesn't require a form to stay on, thus allowing to print from class modules.
Being a global object it does not require to be instantiated.

The usage for most of the properties and actions is the same as for a **PrintPreview** control.
But it has another way to handle events and to call the printing routine.
To handle, events see the property **Events** which returns an **PrinterExEvents** object.

There are two ways of calling the printing routine:

First option: To use the **PrepareReport** event of the **PrinterExEvents** object returned by the **Events** property.

Sample usage in a class module:

```vba
Private WithEvents mPrinterExEvents As PrinterExEvents

Private Sub Class_Initialize()
    Set mPrinterExEvents = PrintPreview.Events
End Sub

Public Sub PrintPreviewMyReport()  ' This is the method that will
be called to show the print preview dialog with the report
    PrintPreview.ShowPreview
End Sub

Private Sub mPrinterExEvents_PrepareReport(Cancel As Boolean)
    MyPrintingProcedure
End Sub

Private Sub MyPrintingProcedure()
    Printer.Print "Hello Word"
    ' continue with the printing code
End Sub
```

Second option: To set the printing procedure using the **SetPrintingProcedure** method.

Sample usage in a class module:

```
Private Sub Class_Initialize()
    PrintPreview.SetPrintingProcedure Me, "MyPrintingProcedure"
End Sub

Public Sub PrintPreviewMyReport()  ' This is the method that will
be called to show the print preview dialog with the report
    PrintPreview.ShowPreview
End Sub

Public Sub MyPrintingProcedure()
    Printer.Print "Hello Word"
    ' continue with the printing code
End Sub
```

Note that the printing routine must be declared as **Public** with this approach.

Also do not forget to add the code necessary to replace the Printer object to a bas module:

```
Public Property Get Printer() As Printer
    Set Printer = BSPrinter.PrinterReplacement
End Property

Public Property Set Printer(nPrinter As Printer)
    Set BSPrinter.PrinterReplacement = nPrinter
End Property
```

<u>Properties</u>:

AllowUserChangeMargins *property*:

Parameters information:
Type: **BSAllowUserChangeMarginsConstants**

Description:

Returns or sets a value that determines whether the user will be able to change the margins from the Page Setup dialog box.

Possible values:
   0 - bsCMNo: The user will not be able to change the margins. The controls for changing margins in the **Page setup dialog box** will be disabled.
   1 - bsCMYes: The user will be able to change the margins. The controls for changing margins in the **Page setup dialog box** will be enabled.
   2 - bsCMWhenHandlingMargins: The user will be able to change the margins if the **HandleMargins** property is set to **True**.

The default setting is **bsCMWhenHandlingMargins**.

AllowUserChangeOrientation *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user will be able to change the page **Orientation** from the **Print Preview and Page Setup dialog boxes**. He will still be able to change it from the **Printer dialog box**. For not displaying the **Printer dialog box**, use the **AllowUserChangePrinter** property (he won't be able to choose the printer device either).

The default setting is **True**.

AllowUserChangePageNumbers *property*:

Parameters information:
Type: **BSAllowUserChangePageNumbersConstants**

Description:

Returns or sets a set of flags that determine whether the user will be able to change the page numbers settings such as **font, position, format and color**, even removing page numbers entirely.

The default setting is **bsCPAllowChangeAll**.

The flags can be used together with the OR operator, such as **bsCPAllowChangeFont** OR **bsCPAllowChangeColor**. In this case the user won't be able to change the position and format.

AllowUserChangePaperSize *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user can change the paper size from the **Page Setup dialog box**. He will still be able to change it from the **Printer dialog box**. For not displaying the **Printer dialog box**, use the **AllowUserChangePrinter** property (he won't be able to choose the printer device either).

The default setting is **True**.

AllowUserChangePrinter *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user will be able to change the Printer in the **Printer dialog box**.
If set to **False**, when the user clicks the Print button at the toolbar, the report will be sent to the printer immediately without displaying the **Printer dialog box**.

The default setting is **True**.

AllowUserChangeScale *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the user can change the scale of the report. If set to **False** the controls that let the user to change the scale will not be visible in the **Print Preview dialog box**.
Changing the scale makes everything in the document smaller or bigger.

The default setting is **True**.

AuxiliaryButtonToolTipText *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the ToolTip text of the optional auxiliary button at the toolbar of the **Print preview dialog**.

AuxiliaryButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a custom button will appear on the top toolbar of the **Print Preview dialog box**.

The default setting is **False**.

This button can be used by the client program to perform an action such as displaying a custom dialog or some other task or configuration.
When it is clicked it raises the **AuxiliaryButtonClick** event.

BottomMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the bottom margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

BottomMarginEffective *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending on the setting of the **MarginUnits** property, which indicates the bottom margin actually applied. This value may be greater than the one set in the **BottomMargin** property, since the margin space can be increased if necessary to print the page numbers or footer.

BottomNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the bottom physical space where the Printer device cannot print. The **BottomMargin** cannot be set smaller than that.

Canceled *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value that indicates if the last **ShowPrinter**, **ShowPageSetup**, **ShowPreview** or **ShowSaveToFile** action was canceled or not.
With **SaveToFile**, it returns **True** if it could not save the file at the location or some other error occurred.

Notes:

PrintDirect can also produce it to be set to True when the printer driver raises an error or the **CancelPrint** argument of the **BeforePrint** event was set to **True**.

Collate *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether multiple copies of the report will be printed collated or not. Collating is performed when the number of copies is greater than 1. The Printer device must support collating, otherwise this setting has not effect.

When the copies of the report are printed collated, the pages are printed in this order: 1, 2, 3, 1, 2, 3, 1, 2, 3. And when they are not collated: 1, 1, 1, 2, 2, 2, 3, 3, 3.

The default is **True**.

This setting can also be changed by the user from the **Printer dialog box**.

ColorSupport *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value that indicates whether the printer device can print in color.

CurrentPageInPreview *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the first page visible in the **Print Preview dialog box**.

Note:

This is not necessarily the page number, but the index or position of the page, because the page number may not be the same because of the value of the property **StartingPageNumber** or because there is more than one document in the report due to a **Printer.EndDoc** in the printing procedure.

Events *property*:

Parameters information:
Type: **Object** (Read only)

Description:

Returns a **PrinterExEvents** object that can be used to receive events from the **IPrinterEx** object's interface.

Sample usage:

```
Private WithEvents mPrinterExEvents As PrinterExEvents

Private Sub Form_Load()
    Set mPrinterExEvents = PrintPreview.Events
End Sub

Private Sub mPrinterExEvents_ReportPrinted()
    MsgBox "Printed"
End S
```

If is only necessary for getting events when the object supplied by the **PrintPreview** global property is used instead of a **PrintPreview** control.
When a **PrintPreview** control is used, it is not necessary because the events are handled by the control itself.
It can be also used when using **PrintPreview** controls though, to have a centralized event handler for all the controls.

Flags *property*:

Parameters information:
Type: **BSPrinterFlagsConstants**

Description:

Returns or sets options for the **Printer dialog box**.

FontWidth *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets the font width of the printer object. The font width is generally chosen by Windows to match the selected font height and provide a standard aspect ratio, however by setting this property you can specify fonts that are narrower or wider than the default font.

FormatButtonVisible *property*:

Parameters information:
Type: **BSAutoYesNoConstants**

Description:

Returns or sets a value that determines whether a Format button will be visible at the top toolbar of the **Print Preview dialog box** to display the **ShowFormatOptions** dialog box.

The default setting is **bsAutoYN**.

bsAutoYN: The Format button will be visible if a Grid was printed using the **PrintGrid** method (* read details below).
bsNo: The Format button will not be visible.
bsYes: The Format button will be visible regardless a Grid was printed or not. From its dialog the user can also configure the report in grayscale.

Notes:

If the Format button is not visible, but **AllowUserChangePageNumbers** is set to **True**, a button to show the **ShowPageNumbersOptions** dialog will appear instead.

This Format button can also be used to display a custom **Format dialog box** by canceling the display by handling the **CancelDisplay** argument in the **BeforeShowingFormatDialog** event and showing a custom dialog instead.

Its icon and ToolTip text can be customized (see **ToolbarButtonPicture** property and **GetLocalizedText** event).

(*) If more that one grid were printed (using the **PrintGrid** method) in the report, if all the grids were printed with the same style (**GridReportStyle**), the format dialog will appear with the grid style selector visible. But if the grids were printed with different styles, the grid style selector will not be visible.

FromPage *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets a value that determines the **first page to be printed**. It will be set in the **From** text box of the **Printer dialog box**.

GDIPlusGraphics *property*:

Parameters information:
Type: **Long** (Read only)

Description:

Returns a handle to a GDI+ Graphics object that can be used to draw with that technology.

For drawing with GDI+, use this Graphics object instead of the one obtained with the **GdipCreateFromHDC** API, because if doing so, there can be problems with transparencies and device printers that use Postscript because the component uses metafiles.

Notes:
Do not make a cache copy of the handle obtained with this property because it changes for every page. As a side note, the same care must be taken with the **Printer.hDC** property of the replacement **Printer** object.

When you finished drawing, do not delete this object with the GdipDeleteGraphics API. It is deleted by the component at a proper moment when it is no longer needed.

GridReportStyles *property*:

Parameters information:
Type: **GridReportStyles** (Read only)

Description:

Collection of **GridReportStyle** that are available.

HandleMargins *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the margins will be handled automatically by the component or not (they will be handled by the client program).

The default setting is **True**.

Note:
When **HandleMargins** is **True**, the **Printer.ScaleWidth** and **Printer.ScaleHeight** properties return the space within the margins.
Also, **Printer.CurrentX** and **Printer.CurrentY** start at the end of the Left and Top margins respectively.

LeftMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the left margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

LeftNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the left physical space where the Printer device cannot print. The **LeftMargin** cannot be set smaller than that.

LetPrintExceedMargin *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the **Printer.Print** and **PrintPreview.PrintEx** methods will be able to print beyond the right margin (in RightToLeft it is the left margin).

Note: in the **PrintEx** method, if the **Alignment** parameter is set, and this property is set to **True**, it has no effect.

MarginsButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to allow the user to enter into a visual mode for configuring the margins will be visible at the top toolbar.

The default setting is **False**.

Notes:

If the property setting **AllowUserChangeMargins** allows the user to change the margins, by pressing this button, the margins can be configured by dragging them.

If the property setting **AllowUserChangeMargins** does not allow the user to change the margins, pressing this button will show the margins but will not allow them to be changed.

If this button is not visible, the user will still be able to enter this visual mode for seeing/changing the margins by means of the menu that appears by right clicking over the top toolbar or by clicking the menu button at the end of the top toolbar (see **MenuButtonVisible** property), in case it is visible.

The **MarginsButtonVisible** toggles the property **ShowMargins True** and **False**.

MarginUnits *property*:

Parameters information:
Type: **BSMarginUnitsConstants**

Description:

Returns or sets a value that determines the units, **Millimeters** or **Inches**, that the program (not for the final user, for the final user see **MarginUnitsForUser**) will use for the property settings related to margins, which are: **LeftMargin**, **TopMargin**, **RightMargin**, **BottomMargin**, **MinLeftMargin**, **MinTopMargin**, **MinRightMargin**, **MinBottomMargin**, **LeftNonPrintableArea**, **TopNonPrintableArea**, **RightNonPrintableArea**, **BottomNonPrintableArea**, **TopMarginEffective** and **BottomMarginEffective**.

The default setting is **vbMillimeters**.

MarginUnitsForUser *property*:

Parameters information:
Type: **BSMarginUnitsForUserConstants**

Description:

Returns or sets a value that determines the units, **Millimeters** or **Inches**, that will be used for the margins in the **Printer dialog box**.

The default setting is **bsMUUserLocale**, that means that it will use Millimeters or Inches based in the Windows locale user setting.

MaxScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the maximum allowed value in percentage that the user can set for the Scale in the Print Preview dialog box.

MenuButtonIcon *property*:

Parameters information:
Type: **BSMenuButtonIconConstants**

Description:

Returns or sets a value that determines the icon of the menu button (at the end of the top toolbar, see **MenuButtonVisible** property).

The default setting is **bsIconStyleVerticalEllipsis**.

MenuButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to display a menu will be visible at the end of the top toolbar.

The default setting is **True**.

Notes:

Even if this property setting is to show the button, the visibility of this button will also depend on the width of the screen, whether there is enough space to display the button or not.

The menu can also be displayed, even if this button is not visible, by right clicking on a free space of the top toolbar.

MinBottomMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum bottom margin that the user can set.

The default value is **0**.

MinLeftMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum left margin that the user can set.

The default value is **0**.

MinRightMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum right margin that the user can set.

The default value is **0**.

MinScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the minimum allowed value in percentage that the user can set for the Scale in the **Print Preview dialog box**.

MinTopMargin *property*:

Parameters information:
Type: **Single**

Description:

Return or set the minimum top margin that the user can set.

The default value is **0**.

PageCount *property*:

Parameters information:
Type: **Long** (Read only)

Description:

Returns the total number of pages.

PageNumbersColor *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the color that will be used to print the page numbers.
This font color setting can also be changed by the final user from the **Page Numbers Options or Format dialog boxes** if **AllowUserChangePageNumbers** is set to **True** (it is the default setting).

The default color is **0 - vbBlack**.

PageNumbersFont *property*:

Parameters information:
Type: **Font**

Description:

Returns or sets the Font that will be used to print the page numbers.
This font setting can also be changed by the user from the **Page Numbers Options or the Format dialog boxes** if **AllowUserChangePageNumbers** is set to **True** (it is the default setting).

The default Font is **Arial 12 Bold**.

PageNumbersFormat *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a string that determines how the page numbers will be formatted.

These characters strings have special meaning, and will be replaced with:
   {#}: page number
   {##}: total page count
   {r}: page number in roman numbers, lowercase
   {Nr}: total page count in roman numbers, lowercase
   {R}: page number in roman numbers, uppercase
   {NR}: total page count in roman numbers, uppercase
   {a}: page number using letters as numbers, lowercase
   {Na}: total page count using letters as numbers, lowercase
   {A}: page number using letters as numbers, uppercase
   {NA}: total page count using letters as numbers, uppercase
   {i}: page number in sequence of i
   {Ni}: total page count in sequence of i
   {I}: page number in sequence of I
   {NI}: total page count in sequence of I

All other characters will be printed verbatim.
But there is one exception: if the full string is "Default". In this case the default formatting will be used, that is "{#}" (just the page number).

Notes:

This setting sets the default format, but the user can change it from the **Page Numbers Options or the Format dialog boxes**, that appear pressing the Page Numbers Options or Format toolbar buttons at the **Print Preview dialog box**. The user will be able to change it as long as the **AllowUserChangePageNumbers** property setting is set to **True** (it is the default setting).

From these dialog boxes the user can select from some predefined formats that are localized, being these formats in English:
"{#}"
"-{#}-"
"{#} / {##}"
"{#}/{##}"
"{#} of {##}"
"{r}"
"{R}"
"{a}"
"{A}"
"pg. {#}"
"pg. {#} / {##}"

"pg. {#}/{##}"
"pg. {#} of {##}"
"Page {#}"
"Page {#} / {##}"
"Page {#}/{##}"
"Page {#} of {##}"
"page {#}"
"page {#} / {##}"
"page {#}/{##}"
"page {#} of {##}"

This list can be changed using the event **GetLocalizedText**, by returning the Text parameter of TextID's:

    bsPPUI_General_PageNumbersFormatStrings_Count
    bsPPUI_General_PageNumbersFormatString_Item + N

For example, when receiving in the **GetLocalizedText** event with **bsPPUI_General_PageNumbersFormatStrings_Count** as argument for **TextID**, the **Text** parameter must return the number of different formats for page numbers. The default is "21" for English **LanguageID** (different languages may have different number of page number formatting options).

After getting that number, the **GetLocalizedText** event will be raised that number of times with **TextID** argument ranging from **bsPPUI_General_PageNumbersFormatString_Item** to **bsPPUI_General_PageNumbersFormatString_Item** + 21 - 1 (or whatever number that was returned before in **bsPPUI_General_PageNumbersFormatStrings_Count**). The **Text** parameter is used to return the format string for each option in that event procedure.

In English the default predefined page numbers format strings available are "{#}" to "page {#} of {##}" listed above.

PageNumbersPosition *property*:

Parameters information:
Type: **BSPageNumbersPositionConstants**

Description:

Returns or sets a value that determines the position of the page numbers.

The default setting is **bsPosDefault**.

Notes:

To set the distance of the page numbers from the bottom of the page or from the top of the page, use **SetFooter** and **SetHeader** methods with the **DistanceFromEdge** argument to the desired value. It is not necessary to set the other parameters.

If you want not to offer some of this position options to the user, you can hide one or more of them by setting the **PageNumbersPositionFlags** property.
If you want to prevent the user from changing the page number format at all, set the **AllowUserChangePageNumbers** property without the flag **bsCPAllowChangeFormat**.

PageNumbersPositionFlags *property*:

Parameters information:
Type: **BSPageNumbersPositionFlagsConstants**

Description:

Returns or sets flags that determine the options that will be available to the user for the position of page numbers.

Note: yo can combine the flags, or make boolean operations with them.
Fort exaple:

```
PrintPreview1.PageNumbersPositionFlags = bsPFAllAvailable And
(Not bsPFAllTopAvailabl
```

will show all positions except the top ones.

PageSetupButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to configure Page Setup will be available on the top toolbar of the **Print Preview dialog box**.

The default setting is **True**.

PageView *property*:

Parameters information:
Type: **BSPageViewConstants**

Description:

Returns or sets a value that determines the page view layout that will be selected by default in the **Print Preview dialog box**.

The default setting is **bsPVFitHeight**.

Note: in case that this property is set to **bsPVNormalSize** and the property **PageViewNormalSizeButtonVisible** is set to **False**, the 'View normal size' button will be displayed anyway at the toolbar.

PageViewNormalSizeButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to display the pages in their normal size will be visible at the top toolbar.

The default setting is **False**.

In case this button is displayed, it will be place first and next to the other view buttons: **View full width**, **View full height** and **View multiple pages**.

PreviewDialogCaption *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the text that will be displayed as the caption of the **Print Preview dialog box**.
If not set, it will show the Product name of the program & " - Print preview " (localized in the available languages).

Notes:

When running in the development environment, the client program is "Visual Basic". But it will show the proper client program name once compiled.

The part of the dialog caption localized as "Print preview" in English has the following **TextID** in the **GetLocalizedText** event:

bsPPUI_PreviewDialog_Caption

PreviewPagesAccuracy *property*:

Parameters information:
Type: **BSPreviewPagesAccuracyConstants**

Description:

Returns or sets the speed and accuracy of the page rendering in the **Print Preview dialog** box.

bsPAFastDrawing: The drawing of the pages in the **Print Preview dialog** box is faster but can be some minor issues with the text scaling in long texts.

bsPAAccurateDrawing: The drawing of the pages in the **Print Preview dialog** box is slower but shows what will be printed without any scaling mismatch. The speed of displaying the pages is fast enough on new machines but can be a bit slow on older machines.

Printed *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if the last operation of **ShowPrinter**, **ShowPreview** or **PrintDirect** resulted in the report printed.

PrintPageNumbers *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines if the component will handle the printing of the page numbers automatically.

The default setting is **True**.

QuickPrintButtonVisible *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether a button to print directly to the default printer or to the printer that was last used if the user settings are being remembered (see the **RememberUserPreferences** property), without showing the **Printer dialog box**, will appear in the top toolbar.

The default setting is **True**.

Note:

The visibility of this button also depends on the width space available, if the space is not enough, this button will not be visible even it this property setting is **True**.

RememberUserPreferences *property*:

Parameters information:
Type: **BSRememberUserPreferencesConstants**

Description:

Returns or sets a value that determines whether the user settings are saved and when they are saved.

These settings include things like scale, paper size, paper orientation, selected printer device, page numbers position, etc.
The default setting is: **3 - bsSPWhenPrinted**.
See **BSRememberUserPreferencesConstants** for details on each value.

Note:

In the **PrintPreview** control, the settings are saved for each control instance, but when the same control is used to print more than one report, the settings can be saved per report. For that it is necessary to set a different **ReportID** for each report. Otherwise, the settings will be saved as a whole, all reports sharing the same last settings.
In the case of using the **IPrinterEx** interface (obtaining the object from the global property **PrintPreview**) instead of a control **PrintPreview**), it is necessary to set a **ReportID** to save user preferences (even if there is only one report), otherwise they will not be saved.

ReportID *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a String that is used time to identify the current report. It is used by the component to store the user's printing preferences for each different report.

If you print more than one report with the same control, use it to identify each report. Otherwise the user's preferences will be remembered as a whole.

Note: For the user's preferences to be remembered, **RememberUserPreferences** property setting must be set to other than **bsSPNever**

ReportName *property*:

Parameters information:
Type: **String**

Description:

Returns or sets a String that is used to identify the print job in the Windows printer spooler.
When ReportName is set, the text of the print job will be "Product name of the client program - " & ReportName. But when it is not set just the Product name of the client program will be displayed.

If this property is set, the report name will also appear as part of the title bar caption of the **Print preview dialog** box.

Notes: When running in the IDE (the development environment), the Product name of the client program will be identified as "Visual Basic". But once it runs compiled, it will be the name of your program.

For example, if the name of the program is "My Program" and ReportName is set to "Report 1", then the print job will be identified as "My Program - Report 1".

RightMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the **MarginUnits** property, that determines the right margin that will be set in new pages. When the current page is other than the last page, it returns the margin value of the current page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't be allowed to change the margins from the dialogs. If you want to set default margins and allow the user to change them, do it before showing the dialog.

RightNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the right physical space where the Printer device cannot print. The **RightMargin** cannot be set smaller than that.

RightToLeftControl *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines if the **RightToLeft** property of the **Printer** replacement object can be set to **True** with no control or only with locales that support right-to-left reading as the original printer does.
The default is **True** to mimic the **VB.Printer** object behavior.

ScalePercent *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the scale factor on which the report will be printed.

Note:

This property is independent from **Printer.Zoom**. Both set an scale, **ScalePercent** is intended for the user, while **Printer.Zoom** can be used by the program to set one or more pages to a different scale. The final scale is defined by both multiplied.

ShowBottomToolbar *property*:

Parameters information:
Type: **BSBooleanUserCanChangeConstants**

Description:

Returns or sets a value that determines whether the **Print Preview dialog box** presents one toolbar at the top and a navigation toolbar at the bottom, or just one toolbar at the top with all the functions accessible from there. Also, if the user will be able to change that arrangement.
The navigation toolbar is used to navigate the different pages.

The default setting is **bsTrueAndUserCanChange**, showing the bottom toolbar.

ShowFirstPageNumber *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the page number, header and footer will be printed when the page number value is 1.

The default setting is **True**

ShowMargins *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether there will be visible lines marking the margins.

The default setting is **False**.

If the **AllowUserChangeMargins** property is set to allow the user to change the margins, the user can also change the margins by dragging these lines.

If the Margins button is visible (see **MarginsButtonVisible** property), the user can toggle on and off the visibility of the margins by pressing that button.
If the button is not visible, the user can still show the margin lines by selecting a menu option.

ShowSketch *property*:

Parameters information:
Type: **BSShowSketchConstants**

Description:

Returns or sets a value that determines whether the Print Preview dialog box, while preparing the report, will show a sketch on the pages or simply empty blank pages.

The default setting is **bsSKWithLargeDocument**.

The sketch will also be displayed when the report is redrawn, for example, when the user changes the paper size, orientation or scale.

SomethingPrintedOnPage *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if there is something printed on the current page.

SomethingPrintedOnReport *property*:

Parameters information:
Type: **Boolean** (Read only)

Description:

Returns a value indicating if there is something printed on the report.

StartingPageNumber *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets the page number to be printed on the first page, or with a negative value on which page the page numbering will start.

The default value is **1**.

Note: If the value is less than 1, then it will set on wich page the page numbers will start. For example, if this property is set to -2, the page numbers will start on the fouth page: Fist page: -2, no page number, second page: -1, no page number either, third page: 0, still no page number, fourth page: 1, the page number is printed here, the page numbering starts on this page.

ToolbarButtonPicture *property*:

Parameters information:
Type: **Picture**

Additional parameter(s):
   In   Required   **ToolbarIconIdentifier**   As
BSToolbarIconIdentifierConstants
   In   Required   **SizeIdentifier**   As
BSIconImageSizeIdentifierConstants

Description:

Returns or sets the images that will be used as the toolbar icons in the Print Preview
dialog box.
They can be bitmap or icon format.
In case of bitmaps, the color for transparency is &HFF00FF or if this color is not present,
the first pixel (at 0,0 position) will be used as the transparent color.

The SizeIdentifier parameter identifies the size that is being set or retrieved.
When changed, normally all 5 images need to be supplied, one at a time.

Note: why 5 sizes of icons? The toolbar buttons have the options of three sizes: small,
medium and big (see **ToolbarIconsSize**).
In addition to that, if the client program is manifested as DPI aware, the component
automatically handles the icon sizes for each DPI setting. It handles these two features
with 5 different icon sizes: 20x20 pixels, 24x24 pixels, 30x30 pixels, 36x36 pixels and
44x44 pixels.

If the client program is not DPI aware, only the 20x20 pixels, 24x24 pixels and 30x30
pixels images are needed.
If **ToolbarIconsSize** is set to a setting where the user can't change the size of the toolbar
buttons, but the program is manifested as DPI aware, only three will be used (depending
on the size if the toolbar buttons, small, medium or big, they can be 20x20/24x24/30x30
for small, 24x24/30x30/36x36 for medium or 30x30/36x36/44x44 for big).
And if the program is not DPI aware and **ToolbarIconsSize** is set to a setting where the
user can't change the toolbar buttons size, only one will be used. (20x20 for small icons,
24x24 for medium or 30x30 for big).

ToolbarIconsSize *property*:

Parameters information:
Type: **BSToolbarIconsSizeConstants**

Description:

Returns or sets a value that determines the size of the toolbar icons at the **Print Preview dialog box**.

The default setting is **bsPPTIconsAutoAndUserCanChange**.
If it is set to any of the settings that end in "UserCanChange", the user can change the size of the toolbar icons by selecting an option of a menu that appears right clicking on the top toolbar or by clicking the menu button, and this setting will be remembered.

ToPage *property*:

Parameters information:
Type: **Long**

Description:

Returns or sets a value that determines the **last page to be printed**. It will be set in the **To** text box of the **Printer dialog box**.

TopMargin *property*:

Parameters information:
Type: **Single**

Description:

Returns or sets a value in millimeters or inches, depending of the setting of the
MarginUnits property, that determines the top margin that will be set in new pages. When
the current page is other than the last page, it returns the margin value of the current
page.
The **default** setting is **25.4 mm or 1 inch**.

Note: if you change margins while the **PrepareReport** event is executing, the user won't
be allowed to change the margins from the dialogs. If you want to set default margins and
allow the user to change them, do it before showing the dialog.

TopMarginEffective *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending on the setting of the **MarginUnits** property, which indicates the top margin actually applied. This value may be greater than the one set in the **TopMargin** property, since the margin space can be increased if necessary to print the page numbers or header.

TopNonPrintableArea *property*:

Parameters information:
Type: **Single** (Read only)

Description:

Returns a value in millimeters or inches, depending of the setting of the **MarginUnits** property, which indicates the top physical space where the Printer device cannot print. The **TopMargin** cannot be set smaller than that.

UIButtonStyle *property*:

Parameters information:
Type: **BSButtonStyleConstants**

Description:

Returns or sets a value that determines the button style of the **Print Preview, Options and Page numbers dialog boxes** buttons and **Print Preview dialog box** toolbars.

The default setting is **bsButtonStyleWindowsStandard**.

UIFontName *property*:

Parameters information:
Type: **String**

Description:

Returns or sets the name of the font that will be used in the custom dialogs.
If not set, it will use the system font setting for Message box dialogs.

UILanguage *property*:

Parameters information:
Type: **Variant**

Description:

Returns or sets a value that determines the **language of the GUI captions** and other texts shown to the final user.
The default setting is 0 (automatic) and it means that the language will be automatically selected between the languages currently supported by the component by using Windows user locale setting. If a language is not supported by the component, the **English** captions will be used.

The client program can supply these captions in other languages not supported by the component by coding in the **GetLocalizedText** event's procedure.
This event can also be used to override (or customize) one or more captions in one or more languages, even when the language is supported by the component.

The constants that identify the language are in the **BSLanguageIDConstants** enumeration, but only some of the languages have built-in captions inside the component. This enumeration is provided in the documentation as a reference, and need to be declared in the client program to be used as named constants.

Note: it can also accept strings like "de" for German or "en-gb" for English UK, but that only works for languages that have built-in support.

UseAltScaleIcons *property*:

Parameters information:
Type: **Boolean**

Description:

Returns or sets a value that determines whether the buttons to increase and decrease the scale that appear on the toolbar of the **Print Preview dialog box** will use alternative icons.
The standard icons are "+" and "-", and the alternative icons add a letter "A" to each icon, being "A+" and "A-".
The default setting is **False**.

Methods:

AddWatermark *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Angle**	As Long
	Default value: 0		
In	Optional	**OpacityPercent**	As Single
	Default value: 15		
In	Optional	**ZPosition**	As
BSWatermarkZPositionConstants			Default value: 1 - bsWPForeground
In	Optional	**Color**	As Long
	Default value: 0		
In	Optional	**PositionUnits**	As
BSWatermarkPositionUnitsConstants			Default value: 0 -
bsPUPercentageOfPage			
In	Optional	**Left**	As Single
	Default value: 15		
In	Optional	**Top**	As Single
	Default value: 20		
In	Optional	**Width**	As Single
	Default value: 70		
In	Optional	**Height**	As Single
	Default value: 60		
In	Optional	**Font**	As Font
	Default value: 0		
In	Optional	**ID**	As String

Description:

Adds a watermark text that will be printed on each page.
The only required parameter is the watermark text.
Any number of watermarks can be added.
The watermarks need to be added from the procedure that prints the report. With the PrintPreview control it means that they must be added when the **PrepareReport** event is executing, that is, from inside that event.

Parameters explained.

Text: the text of the watermark. It may contain more than one line by adding line feeds (vbCrLf).

Angle: the angle in that defines the direction of the watermark, between 0 and 360 degrees. The default is 0 and means horizontal. A value of 45 produces diagonal

watermark and a value of 90 a vertical watermark.

OpacityPercent: it defines the opacity of the watermark. The value must be between 0 and 100. 0 means transparent and 100 means opaque. The default is 15.

ZPosition: defines if the watermark is placed behind the printing of the document or above. The default value is **bsWPForeground** that means above.
Note: When the watermark is placed on the foreground, it uses GDI+ that is available from Windows XP.

Color: sets the fore color of the text. The default value is **vbBlack**

PositionUnits: defines the units used in the following position and size parameters.
    bsPUPercentageOfPage: The watermark position and size are set in percentage of the page (of the full page, without taking into account margins, but taking into account the non-printable area).
    bsPUPrinterScaleMode: The watermark position and size are set in the current ScaleMode units of the Printer object. If HandleMargins is True, the position and size is set inside the margins. Printer.ScaleWidth and Printer.ScaleHeight return the full ranges as a reference.

Left, **Top**, **Width** and **Height**: define the position of the watermark in the page. The default values are: 15, 20, 70 and 60, that along with **PositionUnits** set to **bsPUPercentageOfPage** (also the default value) position the watermark in the middle of the page.

Font: sets the font. The size is automatically changed to fit the space that was defined with the position parameters.

ID: needed only if you are going to call the **RemoveWatermark** method. It is used to identify the watermark. More than one watermark can share the same **ID** (in that case they will be removed together by **RemoveWatermark**.

Note: for printing the watermark in right to left direction, the **Printer** must be set with **RightToLeft** to **True** before adding the watermark.

ClearPage *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Clears anything that has been printed on the current page without removing the page.

The current page can be known from **Printer.Page**.

DeleteAllUserPreferences *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Deletes all the reports user preferences that were previously saved for all the application.

Note:

The custom styles (**GridReportStyle**) that the user saved are not deleted.

You can delete them with this code:

```
Dim Style As GridReportStyle

For Each Style In PrintPreview1.GridReportStyles
    If Style.Custom Then
        PrintPreview1.GridReportStyles.Remove Style
    End If
Next
```

DeleteUserPreferences *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Optional   **ReportID**   As String

Description:

Deletes the user preferences that were previously saved.

For the **PrintPreview** control, if no **ReportID** is set (in this property), the last settings are saved per control instance. If **ReportID** property setting is set, they are saved per control instance + **ReportID**.

The optional parameter **ReportID** can be used to delete the settings of a different report than the one currently set in the **ReportID** property (if any).

Note:

The custom styles (**GridReportStyle**) that the user saved are not deleted.

GetCurrentGridReportStyle *method*:

Parameters information:
Return Type: **GridReportStyle**
No parameters.

Description:

Returns the **GridReportStyle** that will be used to draw the grid(s) on the report.
In the printing procedure it can be used to draw things with the style (colors and other properties) chosen by the user.

GetDocFirstPageIndex *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the page index where a document starts.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page index of the first page of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the index of the first page of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetDocLastPageIndex *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the page index where a document ends.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page index of the last page of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the index of the last page of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetDocPageCount *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
   In   Optional   **ReferencePageIndex**   As Variant

Description:

Returns the number of pages of a document.

Note:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers and page count, and by calling this method it is possible to get the page count of a particular document.
If the **ReferencePageIndex** parameter is supplied, it will return the page count of the document that the page referenced by **ReferencePageIndex** belongs. If this parameter is omitted, the current page will be used as the reference.

GetFooterPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
    In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the footer in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not have a footer, it returns -1.

GetHeaderPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
    In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the header in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not have a header, it returns -1.

GetLastDeviceName *method*:

Parameters information:
Return Type: **String**
No parameters.

Description:

Returns the last device name that is saved as user preference for the current **PrintPreview** control or **ReportID** setting.

GetPageNumber *method*:

Parameters information:
Return Type: **Long**
Parameter(s):
    In   Optional   **PageIndex**   As Variant

Description:

Returns the page number for a given page index. If **PageIndex** is omitted, it returns the page number of the current page.

Notes:

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, and by calling this method it is possible to get the page number of any page. The parameter **PageIndex** must have a number between **1** and **PageCount**.

The first page number can be set to other than 1 using the **StartingPageNumber** property. If there is more than one document in the report, the **StartingPageNumber** setting will be applied only to the first one.

GetPageNumberPos *method*:

Parameters information:
Return Type: **Single**
Parameter(s):
   In   Required   **PosProperty**   As **BSPositionPropertyConstants**

Description:

Returns the position of the page number in the current page.

The parameter **PosProperty** must specify if the left, top, width or height is required.

If the page does not print the page number, it returns -1.

GoToPage *method*:

Parameters information:
Return Type: None.
Parameter(s):
    In   Required   **PageIndex**   As Long

Description:

By calling this method, the current page can be set to any of the already printed pages.

Notes:

If the **PageIndex** argument is set to a non-existent page, an error 9 (subscript out of range) will happen.

If, when printing text on a page, the end of a page is reached, when printing more text, it will be added to the next page.
You can use any of the graphics methods on the page, and if a new page is added by calling **Printer.NewPage**, the new page will be inserted following the current page.

In the case that the object holds more than one document (that can happen if an **Printer.EndDoc** statement was added in the middle of the printing routine), the **PageIndex** argument may not correspond to the page number. That also can happen if the **StartingPageNumber** property is set to other than 1.

PrintDirect *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Sends the report directly to the printer, without displaying the **Print Preview or Printer dialog boxes**.

Note:

If **RememberUserPreferences** is set to other than **bsSPNever**, and the user settings have been saved, it will be printed with the last user selected settings.

PrinterExists *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):
    In   Required   **DeviceName**   As String

Description:

It returns **True** if the printer pointed by **DeviceName** exists, but does not select the printer.

PrintEx *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**Text**	As Variant
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Width**	As Variant
In	Optional	**Height**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
In	Optional	**CancelIfTextExceedsWidth**	As Boolean
	Default value: False		
Out	Optional	**TextExceedsWidth**	As Boolean
Out	Optional	**StartsNewPage**	As Boolean

Description:

Like the **Printer.Print** method, it allows to print text but with some additional options.

Parameters explained:

Text: the text to be printed

Options: can be any of these values:
   bsLineFeed = 0
   bsNoLineFeed = 1
   bsAddTab = 2
The default for this optional parameter is **bsLineFeed**, that means that like the original
Printer.Print method, it will add a line feed automatically after the text.
On the contrary, if this parameter is set to **bsNoLineFeed**, then no line feed will be added.
It is like the function of the semicolon (;) in the original **Printer.Print** method.
bsAddTab adds a Tab after the text, just like the comma (,) in the Printer.Print method.

X, **Y**, **Width** and **Height**: These parameters can define the rectangle on the page where
the text will be printed. If they are omitted, the **Printer.CurrentX** and **Printer.CurrentY** will
be used for the position, and the rectangle is the available space on the page. The text

can wrap to fit in the rectangle.

Alignment: Sets the alignment of the text in the available space. If the rectangle was set by the **X**, **Y**, **Width** and **Height** arguments, the text will be aligned in that rectangle according to the setting of this parameter.
Possible values are:
   bsAlignLeftTop = 0
   bsAlignLeftCenter = 1
   bsAlignLeftBottom = 2
   bsAlignCenterTop = 3
   bsAlignCenterCenter = 4
   bsAlignCenterBottom = 5
   bsAlignRightTop = 6
   bsAlignRightCenter = 7
   bsAlignRightBottom = 8
The default setting for this parameter is **bsAlignLeftTop**

WordWrap: determines whether the text will wrap into another line if it is too long to fit the available horizontal space.
The default setting for this parameter is True.
Note: This is one of the main advantages over the original **Printer.Print** method.

WordBreak: similar to WordWrap, but in this case it determines whether the words can be split into more than one line when the text doesn't fit into the available horizontal space.
The default setting for this parameter is **False**.
Note: the difference between **WordWrap** and **WordBreak**, is that **WordWrap** does not split words. **WordBreak** implies **WordWrap**.

RightToLeftReading: Sets the text direction right-to-left or left-to-right without changing the RightToLeft property setting.
If set to **bsAutoYN** (default setting), the RightToLeft property setting is used instead to set the text direction.
Note: **True** is converted to **bsYes**

RectMarginsX, **RectMarginsY**:
These values can define margins within the rectangle set by the parameters **X**, **Y**, **Width** and **Height**; therefore, they affect the position of the text that is according to the Alignment parameter.
They must be in the current ScaleMode of the Printer.

CancelIfTextExceedsWidth:
Determines whether the printing operation will be aborted if the horizontal space is smaller than required.
The default setting for this parameter is **False**.

TextExceedsWidth: This Boolean parameter needs to be passed **by reference** and **is a return value**. Returns **True** if the required horizontal space is greater than the available space.

StartsNewPage: This Boolean parameter needs to be passed **by reference** and **is a**

return value. Returns **True** if this print operation starts a new page.

PrintForm *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**Form**	As Object
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**OnlyClientArea**	As Boolean
	Default value: False		
In	Optional	**ScaleFactor**	As Single
	Default value: 1		
In	Optional	**CenteredHoriz**	As Boolean
	Default value: False		
In	Optional	**ShrinkToFit**	As Boolean
	Default value: True		
Out	Optional	**PrintedX**	As Variant
Out	Optional	**PrintedY**	As Variant
Out	Optional	**PrintedWidth**	As Variant
Out	Optional	**PrintedHeight**	As Variant

Description:

Prints the image of a form.

It is similar to **PrintWindow**, but the first argument accepts a **Form** object instead of a window handle.

Parameters:

Form: form to be printed.

X: the X position where the image will be printed. If not supplied, CurrentX is used.

Y: the Y position where the image will be printed. If not supplied, CurrentY is used.

OnlyClientArea: If **True**, only the client area will be printed (not the title bar and borders)

ScaleFactor: Scale of the printed image, the default is 1.

PrintedX Returns the X position where the image was printed.

PrintedY Returns the Y position where the image was printed.

PrintedWidth Returns the Width of the printed image.

PrintedWidth Returns the Height of the printed image.

Notes:

It requires Windows XP or later.
Controls that use GDI+ may not appear.

Printer.CurrentX won't be updated, but **Printer.CurrentY** will be positioned after the image.

PrintGrid *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):

In	Required	**Grid**	As Object
In	Optional	**Style**	As Variant
In	Optional	**UserCanChangeStyle**	As Boolean

Default value: True

| In | Optional | **GridPosition** | As **BSGridPositioningConstants** |

Default value: 0 - bsGACentered

| In | Optional | **ColsToHide** | As Variant |
| In | Optional | **PrintCellsFormatting** | As |

BSPrintCellsFormattingConstants   Default value: 0 - bsPCFPrintAll

| In | Optional | **IgnoreEmptyRowsAtTheEnd** | As Boolean |

Default value: True

Description:

Prints the contents a grid.
Supported types are **MSFlexGrid**, **MSHFlexGrid** and compatibles.

Parameters explained:

Grid: A reference to the Grid control to be printed. This is the only parameter that is required.

Style: This optional parameter can set the Style of the Grid. It can be an object **GridReportStyle**, a number identifying a Style or a String with the Name of a Style.
The style can be one built-in, or one that was added by calling the **Add** method of the **GridReportStyles** collection.

UserCanChangeStyle: Determines if the user can customize the look of the printed grid by selecting another Style.
Its default value is **True**.

GridPosition: Determines the position of the grid if it is smaller than the page width.
Possible values are:
    bsGACentered = 0
    bsGALeft = 1
    bsGARight = 2
    bsGAStretch = 3
    bsGAEnlargeLastColumn = 4
    bsGACenteredAndUserCanChange = 8
    bsGALeftAndUserCanChange = 9
    bsGARightAndUserCanChange = 10
    bsGAStretchAndUserCanChange = 11
    bsGAEnlargeLastColumnAndUserCanChange = 12
The default value is **bsGACentered**.

ColsToHide: optional array of Boolean values that determines what columns to hide (not to print).

PrintCellsFormatting: Determines whether the cells formatting will be printed and what.
Possible values are:
    bsPCFPrintAll = 0
    bsPCFIgnoreForeColor = 1
    bsPCFIgnoreBackColor = 2
    bsPCFIgnoreFonts = 4
    bsPCFIgnoreTextAlignment = 8
    bsPCFIgnorePictures = 16
    bsPCFIgnoreAll = 31
The default value is **bsPCFPrintAll**.

IgnoreEmptyRowsAtTheEnd: If there are empty rows at the end of the grid (not filled with data), it determines whether these rows will printed or not.
The default value is **True** (that means that the empty rows at the end will not be printed).

Return value:
When the grid has many rows, the **Print Preview dialog box** will show a window with the progress of loading the data and preparing the preview, and it will show a button that allows to cancel the operation. If the user clicked the button to cancel the operation, the **Print Preview dialog** won't be displayed and this method will return **False**, meaning that the operation was canceled. In a normal operation it returns **True**.

PrintRichTextBox *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Required   **RTBControl**   As Object

Description:

Prints the contents of a RichTextBox control.

The parameter **RTBControl** is a reference to the Rich TextBox control to be printed.

PrintWindow *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**hWnd**	As Long
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**OnlyClientArea**	As Boolean
	Default value: False		
In	Optional	**ScaleFactor**	As Single
	Default value: 1		
In	Optional	**CenteredHoriz**	As Boolean
	Default value: False		
In	Optional	**ShrinkToFit**	As Boolean
	Default value: True		
Out	Optional	**PrintedX**	As Variant
Out	Optional	**PrintedY**	As Variant
Out	Optional	**PrintedWidth**	As Variant
Out	Optional	**PrintedHeight**	As Variant

Description:

Prints the image of a window given its handle.
The window can be a control, a form, or any window that has a DC.
The printed image is how the control or window appears on the screen.
Windowless controls, such as Label or Shape, cannot be printed using this method.
The control must be visible, but it can be placed outside the visible area or covered with other controls.

Parameters:

hWnd: handle of the window to be printed.

X: the X position where the image will be printed. If not supplied, CurrentX is used.

Y: the Y position where the image will be printed. If not supplied, CurrentY is used.

OnlyClientArea: If **True**, only the client area will be printed (not the title bar and in some windows types the borders and scroll bars)

ScaleFactor: Scale of the printed image, the default is 1.

PrintedX Returns the X position where the image was printed.

PrintedY Returns the Y position where the image was printed.

PrintedWidth Returns the Width of the printed image.

PrintedWidth Returns the Height of the printed image.

Notes:

It requires Windows XP or later.

Printer.CurrentX won't be updated, but **Printer.CurrentY** will be positioned after the image.

ReleasePrinterObject *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Releases the Printer object from the current control.

Note: if you used **CapturePrinterObject** for one control and now need to set it to another control, it is not necessary to do a release first (it is done automatically).

RemoveAllWatermarks *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Removes all the watermarks.

Notes:
If you are calling **RemoveAllWatermarks** from within the printing procedure (while in the **PrepareReport** event of the **PrintPreview** control) to remove the watermarks from certain page and ahead, call it before the **Printer.NewPage** that adds that page.
Also, if you are calling **RemoveAllWatermarks** from within the printing procedure, add the watermarks by calling **AddWatermark** method inside the printing procedure too.

RemovePage *method*:

Parameters information:
Return Type: None.
No parameters.

Description:

Removes de current page.

To remove pages that are not the last one, use **GoToPage** method.
The current page can be known from **Printer.Page**.

RemoveWatermark *method*:

Parameters information:
Return Type: None.
Parameter(s):
   In   Required   **ID**   As String

Description:

Removes a watermark.

The **ID** argument identifies the watermark or watermarks to remove.
If there is no watermark with this ID, it does not generate an error.

SaveToFile *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**DeviceNameToUse**	As String
In	Required	**FileFullPath**	As String

Description:

Saves the document to a file instead of printing it.
It should be used only with printers drivers that produce a meaningful output file.
The file format and its corresponding extension depends on the printer driver. Most drivers will produce an intermediate file that is used by the spooling system, such as PostScript files, but some printers driver can produce a useful output such as PDF files or XPS files.
In Windows 10, the "Microsoft Print to PDF" driver produces a PDF file, whereas "Microsoft XPS Document Writer" outputs XPS files.

Parameters explained:

DeviceNameToUse is the printer driver to be used.

FileFullPath is the full path of the output file.

Notes: Unlike the **ShowSaveToFile** method, this method saves the file without displaying any dialog unless the driver itself shows dialogs on the operation.
If the printer driver is not installed the end user will get a message telling that the file cannot be saved because the printer driver is not installed.
You can use the **PrinterExists** method to check whether a particular printer driver is installed.

SelectPrinter *method*:

Parameters information:
Return Type: **Boolean**
Parameter(s):
    In   Required   **DeviceName**   As String
    In   Optional   **PreserveUserPrintingPreferences**   As Boolean
        Default value: False

Description:

It selects a printer.

Parameters explained:

DeviceName: device to select.

PreserveUserPrintingPreferences: if **True**, the current printing settings such as Orientation, PaperSize, ColorMode will be preserved is the driver supports them. If the paper size is not supported by the printer, it will select the most similar paper. If **False** the settings will be printer default settings. The default value is **False**

It returns **True** if the printer was successfully selected, **False** otherwise.
The reason why a printer selection can fail is because the printer doesn't exists or because **DeviceName** was misspelled.

Note: it is similar to the code:

```
Dim prn As Printer

For Each prn In Printers
    If prn.DeviceName = DeviceName Then
        Set Printer = prn
        Exit For
    End If
Next
```

SetFooter *method*:

Parameters information:
Return Type: None.
Parameter(s):

| In | Optional | **TextOrPicture** | As Variant |
| In | Optional | **Position** | As |

BSHorizontalPositionConstants   Default value: 0 - bsHPosLeft

| In | Optional | **DistanceFromEdge** | As Single |

          Default value: 15

| In | Optional | **MultilineTextAlignment** | As |

BSMultilineTextAlignmentConstants   Default value: -1 -
bsMTAlignAccordingToPosition

| In | Optional | **Font** | As Variant |
| In | Optional | **TextColor** | As Long |

          Default value: 0

| In | Optional | **PictureScale** | As Single |

          Default value: 1

Description:

Sets an image or text that will be printed as a footer on each page, such as a logo or a note.
It can be an image or a text, but not both. This setting is for all the report, it cannot be set for individual pages.

Note:

This feature is not available in the demo version.

Parameters explained:

TextOrPicture: this Variant parameter can accept a stdPicture with an image or a text.
To remove a footer previously set, omit this parameter, or provide an empty Variant, an Object variable containing Nothing or a null String.

Position: sets the position of the image or text (single line text) to the left, right or horizontally centered.
    bsHPosLeft = 0
    bsHPosRight = 1
    bsHPosCenter = 2
The default value is **bsHPosLeft**.

DistanceFromEdge: sets the distance from the edge of the page to the footer.
Note that the footer and header are usually within the margin space.
The default value is **15 millimeters (0.59 inch)**.

MultilineTextAlignment: sets the alignment of the different lines of text when the text does not fit in one line or has vbCrLf in it.

Font : sets the font for the text.
Default font is **Arial 12 Regular**.

TextColor: sets the fore color of the text.
The default color is **vbBlack**.

PictureScale: sets the scale of the picture.
The default scale is **1**.

SetHeader *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Optional	**TextOrPicture**	As Variant

In   Optional   **Position**   As
BSHorizontalPositionConstants   Default value: 0 - bsHPosLeft

     In   Optional   **DistanceFromEdge**   As Single
         Default value: 15

     In   Optional   **MultilineTextAlignment**   As
BSMultilineTextAlignmentConstants   Default value: -1 -
bsMTAlignAccordingToPosition

     In   Optional   **Font**   As Variant
     In   Optional   **TextColor**   As Long
         Default value: 0

     In   Optional   **PictureScale**   As Single
         Default value: 1

Description:

Sets an image or text that will be printed as a header on each page, such as a logo or a note.
It can be an image or a text, but not both. This setting is for all the report, it cannot be set for individual pages.

Parameters explained:

TextOrPicture: this Variant parameter can accept a stdPicture with an image or a text.
To remove a header previously set, omit this parameter, or provide an empty Variant, an Object variable containing Nothing or a null String.

Position: sets the position of the image or text (single line text) to the left, right or horizontally centered.
     bsHPosLeft = 0
     bsHPosRight = 1
     bsHPosCenter = 2
The default value is **bsHPosLeft**.

DistanceFromEdge: sets the distance from the edge of the page to the footer.
Note that the footer and header are usually within the margin space.
The default value is **15 millimeters (0.59 inch)**.

MultilineTextAlignment: sets the alignment of the different lines of text when the text does not fit in one line or has vbCrLf in it.

Font : sets the font for the text.
Default font is **Arial 12 Regular**.

TextColor: sets the fore color of the text.

The default color is **vbBlack**.

PictureScale: sets the scale of the picture.
The default scale is **1**.

SetPrintingProcedure *method*:

Parameters information:
Return Type: None.
Parameter(s):

In	Required	**Object**	As Object
In	Required	**ProcedureName**	As String
In	Optional	**ParametersArray**	As Variant

Description:

Using the **IPrinterEx** object instead of a **PrintPreview** control is possible to has print preview capabilities from a **Class module** where no controls can be placed, or even from forms if you prefer.
You will still need to place one **PrintPreview** control in a form that loads first to set the license information.

The **IPrinterEx** is the interface of a global object that can be obtained with the **PrintPreview** global property.
Using this object, there are two ways to call **the printing procedure**, one is using events, for that see **Events** property. The other is to set **the printing procedure** with this method.

Parameters explained:

Object: this must be a reference to the class module, form or object where **the printing procedure** is.

ProcedureName: the name of **the printing procedure**. This procedure **must be Public**.

ParametersArray: if **the printing procedure** has parameters, they can come in an array here.

Example without parameters:

```
Private Sub Class_Initialize()
    PrintPreview.SetPrintingProcedure Me, "MyPrintingProcedure"
End Sub

Public Sub PrintPreviewMyReport()  ' This is the method that will
be called to show the print preview dialog with the report
    PrintPreview.ShowPreview
End Sub

Public Sub MyPrintingProcedure()
    Printer.Print "Hello Word"
     ' continue with the printing code
End Sub
```

Example with parameters:

```
Public Sub PrintPreviewMyReport1()
    PrintPreview.SetPrintingProcedure Me, "MyPrintingProcedure",
Array(1, 14)  ' ReportNumber = 1, TitleFontSize = 14
    PrintPreview.ShowPreview
End Sub

Public Sub PrintPreviewMyReport2()
    PrintPreview.SetPrintingProcedure Me, "MyPrintingProcedure",
Array(2, 18)  ' ReportNumber = 1, TitleFontSize = 14
    PrintPreview.ShowPreview
End Sub

Public Sub MyPrintingProcedure(ReportNumber As Long,
TitleFontSize As Long)
    If ReportNumber = 1 Then
        Printer.FontSize = TitleFontSize
        Printer.Print "Report 1"
        Printer.FontSize = 12
        Printer.Print "Hello Word"
         ' continue with the printing code
    Else
        Printer.FontSize = TitleFontSize
        Printer.Print "Report 2"
        Printer.FontSize = 12
        Printer.Print "Hello Word"
         ' continue with the printing code
    End If
End Sub
```

Of course you can send the parameters in other ways in your program, this is just a
possibility.

ShowFormatOptions *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Format dialog box**, where the user can select page numbers options, color or grayscale for the report, and if there is printed a grid, the layout and color of it.

The dialog **ShowPageNumbersOptions** is similar, but it just handles the page numbers options.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPageNumbersOptions *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Page Numbers dialog box**, where the user can select the page numbers options.

The dialog **ShowFormatOptions** is similar, but it handles a few more options, including grid layouts if there is a grid printed in the report.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPageSetup *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Page Setup dialog box**, where the user can select the paper size, page orientation and margins.

If the user changed something and clicked the OK button, it returns **bsDialogResultOK**, otherwise it returns **bsDialogResultCancel**.
Note: If the user did not change anything, even if pressed the OK button, the return value will be **bsDialogResultCancel**.

ShowPreview *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
No parameters.

Description:

Displays the **Print Preview dialog box**.

If the user didn't print the report, the return value will be **bsDialogResultCancel**.

It returns **bsDialogResultOK** if the user printed the report or **bsDialogResultCancel** otherwise.

ShowPrinter *method*:

Parameters information:
Return Type: **BSShowPrinterDialogResultConstants**
No parameters.

Description:

Displays the **Printer dialog box**, where the user can select the printer and other options.

If the user clicked the Print button, it returns **bsPDResultPrint**, if cancelled it returns **bsPDResultCancel**, and if the user clicked the Apply button **bsPDResultApply**.

ShowSaveToFile *method*:

Parameters information:
Return Type: **BSGeneralDialogResultConstants**
Parameter(s):

In	Required	**DeviceNameToUse**	As String
In	Required	**DefaultFileNameOrPath**	As String
In	Optional	**RememberLastPath**	As Boolean
	Default value: True		
In	Optional	**OverwritePrompt**	As Boolean
	Default value: True		

Description:

It saves the document to a file instead of printing it.
It is similar to the **SaveToFile** method, but in this case the user is asked where to save the file.

It should be used only with printers drivers that produce a meaningful output file.
The file format and its corresponding extension depends on the printer driver. Most drivers will produce an intermediate file that is used by the spooling system, such as PostScript files, but some printers driver can produce a useful output such as PDF files or XPS files.
In Windows 10, the "Microsoft Print to PDF" driver produces a PDF file, whereas "Microsoft XPS Document Writer" outputs XPS files.

Parameters explained:

DeviceNameToUse is the printer driver to be used.

FileFullPath is the full path of the output file.

Notes: Some printer drivers might show additional dialogs on the operation.
If the printer driver is not installed the end user will get a message telling that the file cannot be saved because the printer driver is not installed.
You can use the **PrinterExists** method to check whether a particular printer driver is installed.

TextHeightEx *method*:

Parameters information:
Return Type: **Single**
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Width**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
Out	Optional	**StartsNewPage**	As Boolean

Description:

Similar to Printer.TextHeight but with additional parameters.
Usually used in combination with **PrintEx** method. Please refer to **PrintEx** to read the explanation of the parameters.

TextWidthEx *method*:

Parameters information:
Return Type: **Single**
Parameter(s):

In	Required	**Text**	As String
In	Optional	**Options**	As **BSPrintExOptionsConstants**
	Default value: 0 - bsLineFeed		
In	Optional	**X**	As Variant
In	Optional	**Y**	As Variant
In	Optional	**Height**	As Variant
In	Optional	**Alignment**	As **BSAlignmentConstants**
	Default value: 0 - bsAlignLeftTop		
In	Optional	**WordWrap**	As Boolean
	Default value: True		
In	Optional	**WordBreak**	As Boolean
	Default value: False		
In	Optional	**RightToLeftReading**	As **BSAutoYesNoConstants**
	Default value: 2 - bsAutoYN		
In	Optional	**RectMarginsX**	As Single
	Default value: 0		
In	Optional	**RectMarginsY**	As Single
	Default value: 0		
Out	Optional	**StartsNewPage**	As Boolean

Description:

Similar to Printer.TextWidth but with additional parameters.
Usually used in combination with **PrintEx** method. Please refer to **PrintEx** to read the explanation of the parameters.

PrinterExEvents object

Auxiliary object used to receive events from a IPrinterEx object.

Sample usage in a Form:

```vb
Private WithEvents mPrinterExEvents As PrinterExEvents

Private Sub Form_Load()
    Set mPrinterExEvents = PrintPreview.Events
End Sub

Private Sub mPrinterExEvents_ReportPrinted()
    MsgBox "Print done!"
End Sub
```

This is only the code for handling the events, for a working sample add a **PrintPreview** control and a CommandButton to a form and paste this code:

```vb
Private WithEvents mPrinterExEvents As PrinterExEvents

Private Sub Command1_Click()
    PrintPreview1.ShowPreview
End Sub

Private Sub Form_Load()
    Set mPrinterExEvents = PrintPreview.Events
End Sub

Private Sub mPrinterExEvents_ReportPrinted()
    MsgBox "Print done!"
End Sub

Private Sub PrintPreview1_PrepareReport(Cancel As Boolean)
    Printer.Print "Hello world."
End Sub
```

Sample usage in a Class module:

```vb
Private WithEvents mPrinterExEvents As PrinterExEvents

Private Sub Class_Initialize()
    Set mPrinterExEvents = PrintPreview.Events
End Sub

Private Sub mPrinterExEvents_ReportPrinted()
```

```vb
        MsgBox "Print done!"
    End Sub
```

This is only the code for handling the events, for a working sample add this code to a Class module:

```vb
    Private WithEvents mPrinterExEvents As PrinterExEvents

    Private Sub Class_Initialize()
        Set mPrinterExEvents = PrintPreview.Events
    End Sub

    Public Sub PrintPreviewMyReport()   ' This is the method that will
    be called to show the print preview dialog with the report
        PrintPreview.ShowPreview
    End Sub

    Private Sub mPrinterExEvents_PrepareReport(Cancel As Boolean)
        MyPrintingRoutine
    End Sub

    Private Sub mPrinterExEvents_ReportPrinted()
        MsgBox "Print done!"
    End Sub

    Private Sub MyPrintingRoutine()
        Printer.Print "Hello Word"
        ' continue with the printing code
    End Sub
```

And add a CommandButton and this code to a form:

```vb
    Private Sub Command1_Click()
        Dim c As New Class1

        c.PrintPreviewMyReport
    End Sub
```

Events:

AfterShowingPageSetupDialog *event*:

Parameters information:
No parameters.

Description:

Occurs after the **Page Setup dialog box** is displayed. Check the **Canceled** property to see whether the user changed something or not.

AfterShowingPrinterDialog *event*:

Parameters information:
No parameters.

Description:

Occurs after the **Printer dialog box** is displayed. Check the **Canceled** property to see whether the user changed something or not, and the **Printed** property to see whether the report was printed.

AuxiliaryButtonClick *event:*

Parameters information:
Parameter(s):
    Returns value   **UpdateReport**   As Boolean

Description:

Occurs when the user clicks on the Auxiliary button.

It can be used to perform certain action such as displaying a custom dialog or some other task or configuration.

UpdateReport is a return value.
The default value for the UpdateReport is True.
If it is returned with **False the report will not be redrawn, otherwise the PrepareReport** event will be raised and the pages at the **Print preview dialog** will be updated.

BeforePrint *event*:

Parameters information:
Parameter(s):
   Returns value   **CancelPrint**   As Boolean

Description:

Occurs when a report is about to be sent to the printer.
The print action can be canceled by setting the **CancelPrint** argument to **True**.

BeforePrintingPageNumbers *event*:

Parameters information:
Parameter(s):
   Returns value   **Cancel**   As Boolean

Description:

Occurs when the page numbers are about to be printed.
The page numbers printing can be canceled at this time by setting the **Cancel** argument to **True**.

BeforeShowingFormatDialog *event*:

Parameters information:
Parameter(s):
   Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Format Options dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed and you can show your own dialog instead.

BeforeShowingPageSetupDialog *event*:

Parameters information:
Parameter(s):
   Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Page Setup dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed.

BeforeShowingPrinterDialog *event*:

Parameters information:
Parameter(s):
    Returns value   **CancelDisplay**   As Boolean

Description:

Occurs when the **Printer dialog box** is about to be displayed.
If the **CancelDisplay** parameter is set to **True**, it will not be displayed.

CurrentPrinterDefaultSettingChange {PrinterExEvents object} *event*:

Parameters information:
Parameter(s):
    Info   **PropertyName**   As String
    Info   **NewValue**   As Variant

Description:

Occurs when a default setting is changed in Windows for the currently selected printer device (such as default Orientation or Paper Size).

For example if the Orientation changes from vertical to horizontal, the **PropertyName** argument will be "Orientation" and **NewValue** will be 2 - vbPRDPHorizontal

DefaultPrinterChange {PrinterExEvents object} *event:*

Parameters information:
No parameters.

Description:

Occurs when the default printer device is changed in Windows.

EndDoc *event*:

Parameters information:
Parameter(s):

Info	**FirstPageIndex**	As Long
Info	**LastPageIndex**	As Long

Description:

Occurs when the report was prepared but before the page numbers are printed.

GetLocalizedText *event*:

Parameters information:
Parameter(s):

Info	**LanguageID**	As Long
Info	**SubLanguageID**	As Long
Info	**TextID**	As Long
Returns value	**Text**	As String
Info	**Number**	As Long

Description:

Occurs for every caption or text that the component needs to display.

From the procedure of this event it is possible to translate the texts into any language that does not have built-in support by the component.

The **LanguageID** argument identifies the language. If the **UILanguage** property is set to **0 - bsLang_AUTO_SELECT** (default setting), the **LanguageID** argument will have the code number of the actual user locale, but if that property is set to a specific locale code number, it will come always with that number. The list of possible values are in the **BSLanguageIDConstants** enumeration.

SubLanguageID: some languages can have a sub-language ID, it will be according to the locale of the user.
SUBLANG_ENGLISH_UK = 2
SUBLANG_PORTUGUESE_BRAZILIAN = 1
(the above two are already supported without doing anything)
SubLanguageID is intended to be used together with the language **LanguageID**.
You can ignore the **SubLanguageID** parameter if you want to provide the same **Text** for all locations of the language.

The **TextID** argument identifies what caption is requested. The list of possible values with the corresponding texts in English are in the **BSUserInterfaceTextsIDConstants** enumeration.

The way to customize a text is to assign it to the **Text** parameter. It comes with a default text but it is a return parameter.

Number: This parameter is reserved, currently not used.

Notes:

This event can be used to customize the texts, even if the language has built-in support for that language by the component.

For languages that do not have built-in support, English captions will be used.

When translating to a new language, many of them are standard texts that can be found with **https://www.microsoft.com/en-us/language/search**

OrientationChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the user changes the page Orientation.

PaperSizeChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the user changes the paper size.

PrepareReport *event*:

Parameters information:
Parameter(s):
    Returns value   **Cancel**   As Boolean

Description:

Event to call the code to print the report using the **Printer** object replacement. The report must be printed on the **Printer** object replacement from within this event procedure.
This is the main event.

ReportPrinted *event*:

Parameters information:
No parameters.

Description:

Occurs after a report is sent to a printer device.

ScaleChange *event*:

Parameters information:
No parameters.

Description:

Occurs when the scale is changed by the user from the **Print Preview dialog box**.

StartDoc *event*:

Parameters information:
Parameter(s):
   Info   **FirstPageIndex**   As Long

Description:

Occurs when in a document is added the first page.

A report can have more than one document.
If in the middle of the Printing routine there is an **EndDoc** statement, the **Print Preview dialog box** will show more than one document. Each document will have its page numbers, the **FirstPageIndex** argument has the page index of the first page of the current document.

StartPage *event*:

Parameters information:
No parameters.

Description:

Occurs every time a new page is added to the report.

Constants

BSAlignmentConstants enumeration

Constants used to align text in a rectangle.

bsAlignLeftTop = 0
 The text is aligned left, top.

bsAlignLeftCenter = 1
 The text is aligned left, center.

bsAlignLeftBottom = 2
 The text is aligned left, bottom.

bsAlignCenterTop = 3
 The text is aligned center, top.

bsAlignCenterCenter = 4
 The text is aligned center, center.

bsAlignCenterBottom = 5
 The text is aligned center, bottom.

bsAlignRightTop = 6
 The text is aligned right, top.

bsAlignRightCenter = 7
 The text is aligned right, center.

bsAlignRightBottom = 8
 The text is aligned right, bottom.

bsAlignGeneral = 9
 If the text contains a number, it is aligned right, center; otherwise it is aligned left, center.

BSAllowUserChangeMarginsConstants enumeration

Constants used by the **AllowUserChangeMargins** property.

bsCMNo = 0

> The user will not be able to change the margins. The controls for changing margins in the Page setup dialog box will be disabled.

bsCMWhenHandlingMargins = 2

> The user will be able to change the margins if the **HandleMargins** property is set to True.
> This is the default setting of the **AllowUserChangeMargins** property.

bsCMYes = 1

> The user will be able to change the margins. The controls for changing margins in the Page setup dialog box will be enabled.

BSAllowUserChangePageNumbersConstants enumeration

Constants to set which attributes the user will be able to customize or not.

bsCPAllowChangeAll = 15
>   This constant is the sum of the other flags: **bsCPAllowChangeFont** OR **bsCPAllowChangePosition** OR **bsCPAllowChangeFormat** OR **bsCPAllowChangeColor**. The user will be able to change all: font, position, format and color.

bsCPAllowChangeColor = 8
>   Determines whether the user will be able to change the color of the page numbers. For this flag to work, **bsCPAllowChangeFont** flag must also be set.

bsCPAllowChangeFont = 1
>   Determines whether the user will be able to change the font of the page numbers.

bsCPAllowChangeFormat = 4
>   Determines whether the user will be able to change the format of the page numbers.

bsCPAllowChangeNone = 0
>   The user won't be able to change any: font, position, format or color.

bsCPAllowChangePosition = 2
>   Determines whether the user will be able to change the position of the page numbers.

BSAutoYesNoConstants enumeration

Constants used for Boolean like settings that also allows an automatic option.

bsAutoYN = 2
> Automatic Yes or No depending on some factor or circumstance.

bsNo = 0
> Similar to Boolean False.

bsYes = 1
> Similar to Boolean True.

BSBooleanUserCanChangeConstants enumeration

These constants express a Boolean setting but also if the user will be able to change it or not.

bsFalse = 0
False and the user will not have the option to change it.

bsFalseAndUserCanChange = 2
False by default but the user will be able to change it.

bsTrue = 1
True and the user will not have the option to change it.

bsTrueAndUserCanChange = 3
True by default but the user will be able to change it.

BSButtonStyleConstants enumeration

Constants used by the **UIButtonStyle** property.

bsButtonStyleWindowsStandard = 0
    "Windows Standard" button style.

bsButtonStyleShinyMetal = 1
    "Shiny Metal" button style.

bsButtonStyleIlluminated = 2
    "Illuminated" button style.

bsButtonStyle3DClassic = 3
    "3D Classic" button style.

bsButtonStyleSimple2D = 4
    "Simple 2D" button style.

bsButtonStyleSimileAero = 5
    "Simile Aero" button style.

bsButtonStyleLightsAndShadows = 6
    "Lights & Shadows" button style.

bsButtonStyleOrange = 7
    "Orange" button style.

bsButtonStyleBluish = 8
    "Bluish" button style.

bsButtonStylePolished = 9
    "Polished" button style.

bsButtonStyleBlueMoon = 10
    "Blue Moon" button style.

bsButtonStyleWhiteBlue = 11
    "White Blue" button style.

bsButtonStyleWhiteBlueInverted = 12
    "White Blue Inverted" button style.

BSGeneralDialogResultConstants enumeration

Constants used for the return value of dialog boxes **ShowFormatOptions**, **ShowPageNumbersOptions**, **ShowPageSetup** and **ShowPreview**.

bsDialogResultCancel = 0
The user pressed the Cancel button or pressed the OK button but without changing anything or without doing the intended action.

bsDialogResultOK = 1
The user changed something and pressed the OK button.

BSGridPositioningConstants enumeration

Constants to place and size a grid in a report when the available horizontal space is larger than the width of the grid.

bsGACentered = 0
> The grid will be centered horizontally on the page.

bsGALeft = 1
> The grid will be aligned to the left side.

bsGARight = 2
> The grid will be aligned to the right side.

bsGAStretch = 3
> The grid will extend to all available horizontal space on the page.

bsGAEnlargeLastColumn = 4
> The last column will be extended to cover the available horizontal space left on the page.

bsGACenteredAndUserCanChange = 8
> The default position is the same as **bsGACentered** but the user is allowed to choose another position in the Format Options dialog box.

bsGALeftAndUserCanChange = 9
> The default position is the same as **bsGALeft** but the user is allowed to choose another position in the Format Options dialog box.

bsGARightAndUserCanChange = 10
> The default position is the same as **bsGARight** but the user is allowed to choose another position in the Format Options dialog box.

bsGAStretchAndUserCanChange = 11
> The default position is the same as **bsGAStretch** but the user is allowed to choose another position in the Format Options dialog box.

bsGAEnlargeLastColumnAndUserCanChange = 12
> The default position is the same as **bsGAEnlargeLastColumn** but the user is allowed to choose another position in the Format Options dialog box.

BSHorizontalPositionConstants enumeration

Constants for setting the horizontal position of an element.

bsHPosCenter = 2
> The item will be centered.

bsHPosLeft = 0
> The position will be set to the left.

bsHPosRight = 1
> The position will be set to the right.

BSIconImageSizeIdentifierConstants enumeration

Constants for identifying the image size of an icon between 5 sizes.

bs20x20pix = 0
>   Image size 20x20 pixels.

bs24x24pix = 1
>   Image size 24x24 pixels.

bs30x30pix = 2
>   Image size 30x30 pixels.

bs36x36pix = 3
>   Image size 36x36 pixels.

bs44x44pix = 4
>   Image size 44x44 pixels.

BSLanguageIDAllConstants enumeration

Identifiers for the different user locale languages. The list is comprehensive, but only some of them have built-in support in the component.

Note: This enum is not provided by the component. You will need to declare the constants in your code to use them.

bsLang_AUTO_SELECT = 0
>   The language will be selected automatically according to the user locale.

bsLang_ARABIC = 1
>   Arabic.

bsLang_BULGARIAN = 2
>   Bulgarian.

bsLang_CATALAN = 3
>   Catalan.

bsLang_CHINESE_SIMPLIFIED = 4
>   Chinese simplified.

bsLang_CZECH = 5
>   Czech.

bsLang_DANISH = 6
>   Danish.

bsLang_GERMAN = 7
>   German.

bsLang_GREEK = 8
>   Greek.

bsLang_ENGLISH = 9
>   English.

bsLang_SPANISH = 10
>   Spanish.

bsLang_FINNISH = 11
>   Finnish.

bsLang_FRENCH = 12
>   French.

bsLang_HEBREW = 13
>   Hebrew.

bsLang_HUNGARIAN = 14
>   Hungarian.

bsLang_ICELANDIC = 15

Icelandic.

bsLang_ITALIAN = 16
Italian.

bsLang_JAPANESE = 17
Japanese.

bsLang_KOREAN = 18
Korean.

bsLang_DUTCH = 19
Dutch.

bsLang_NORWEGIAN = 20
Norwegian.

bsLang_POLISH = 21
Polish.

bsLang_PORTUGUESE = 22
Portuguese.

bsLang_ROMANSH = 23
Romansh.

bsLang_ROMANIAN = 24
Romanian.

bsLang_RUSSIAN = 25
Russian.

bsLang_BOSNIAN_SERBIAN_CROATIAN = 26
Bosnian, Serbian or Croatian.

bsLang_SLOVAK = 27
Slovak.

bsLang_ALBANIAN = 28
Albanian.

bsLang_SWEDISH = 29
Swedish.

bsLang_THAI = 30
Thai.

bsLang_TURKISH = 31
Turkish.

bsLang_URDU = 32
Urdu.

bsLang_INDONESIAN = 33
Indonesian.

bsLang_UKRAINIAN = 34
Ukrainian.

bsLang_BELARUSIAN = 35

Belarusian.

bsLang_SLOVENIAN = 36
Slovenian.

bsLang_ESTONIAN = 37
Estonian.

bsLang_LATVIAN = 38
Latvian.

bsLang_LITHUANIAN = 39
Lithuanian.

bsLang_TAJIK = 40
Tajik.

bsLang_FARSI_PERSIAN = 41
Farsi or Persian.

bsLang_VIETNAMESE = 42
Vietnamese.

bsLang_ARMENIAN = 43
Armenian.

bsLang_AZERI = 44
Azeri.

bsLang_BASQUE = 45
Basque.

bsLang_SORBIAN = 46
Lower or Upper Sorbian.

bsLang_MACEDONIAN = 47
Macedonian.

bsLang_TSWANA = 50
Tswana.

bsLang_XHOSA = 52
Xhosa.

bsLang_ZULU = 53
Zulu.

bsLang_AFRIKAANS = 54
Afrikaans.

bsLang_GEORGIAN = 55
Georgian.

bsLang_FAEROESE = 56
Faeroese.

bsLang_HINDI = 57
Hindi.

bsLang_MALTESE = 58

Maltese.

bsLang_SAMI = 59
Sami.

bsLang_IRISH = 60
Irish.

bsLang_MALAY = 62
Malay.

bsLang_KAZAK = 63
Kazak.

bsLang_KYRGYZ = 64
Kyrgyz.

bsLang_SWAHILI = 65
Swahili.

bsLang_TURKMEN = 66
Turkmen.

bsLang_UZBEK = 67
Uzbek.

bsLang_TATAR = 68
Tatar.

bsLang_BENGALI = 69
Bengali.

bsLang_PUNJABI = 70
Punjabi.

bsLang_GUJARATI = 71
Gujarati.

bsLang_ORIYA = 72
Oriya.

bsLang_TAMIL = 73
Tamil.

bsLang_TELUGU = 74
Telegu.

bsLang_KANNADA = 75
Kannada.

bsLang_MALAYALAM = 76
Malayalam.

bsLang_ASSAMESE = 77
Assamese.

bsLang_MARATHI = 78
Marathi.

bsLang_SANSKRIT = 79

Sanskrit.

bsLang_MONGOLIAN = 80
Mongolian.

bsLang_TIBETAN = 81
Tibetan.

bsLang_WELSH = 82
Welsh.

bsLang_KHMER = 83
Khmer.

bsLang_LAO = 84
Lao.

bsLang_GALICIAN = 86
Galician.

bsLang_KONKANI = 87
Konkani.

bsLang_MANIPURI = 88
Manipuri.

bsLang_SINDHI = 89
Sindhi.

bsLang_SYRIAC = 90
Syriac.

bsLang_SINHALESE = 91
Sinhalese.

bsLang_INUKTITUT = 93
Inuktitut.

bsLang_AMHARIC = 94
Amharic.

bsLang_TAMAZIGHT = 95
Tamazight, Berber or Amazigh.

bsLang_KASHMIRI = 96
Kashmiri or Koshur.

bsLang_NEPALI = 97
Nepali.

bsLang_FRISIAN = 98
Frisian.

bsLang_PASHTO = 99
Pashto.

bsLang_FILIPINO = 100
Filipino.

bsLang_DIVEHI = 101

Divehi.

bsLang_HAUSA = 104
Hausa.

bsLang_YORUBA = 106
Yoruba.

bsLang_QUECHUA = 107
Quechua.

bsLang_SOTHO = 108
Sotho.

bsLang_BASHKIR = 109
Bashkir.

bsLang_LUXEMBOURGISH = 110
Luxembourgish.

bsLang_GREENLANDIC = 111
Greenlandic.

bsLang_IGBO = 112
Igbo or Ibo.

bsLang_TIGRIGNA = 115
Tigrigna or Tigrinya.

bsLang_YI = 120
Yi, Nuosu, Nosu, Northern Yi, Liangshan Yi or Sichuan Yi.

bsLang_MAPUDUNGUN = 122
Mapudungun or Mapuche.

bsLang_MOHAWK = 124
Mohawk.

bsLang_BRETON = 126
Breton.

bsLang_UIGHUR = 128
Uighur or Uyghur.

bsLang_MAORI = 129
Maori or Te Reo.

bsLang_OCCITAN = 130
Occitan.

bsLang_CORSICAN = 131
Corsican or Lingua Corsa.

bsLang_ALSATIAN = 132
Alsatian or Alsatian German.

bsLang_YAKUT = 133
Yakut, Sakha, Saqa or Saxa.

bsLang_KICHE = 134

Kiche.

bsLang_KINYARWANDA = 135
Kinyarwanda or Urufumbira.

bsLang_WOLOF = 136
Wolof.

bsLang_DARI = 140
Dari or Dari Persian.

bsLang_SCOTTISH_GAELIC = 145
Scottish Gaelic or Scots Gaelic.

bsLang_BOSNIAN_NEUTRAL = 30746
Bosnian Neutral

bsLang_CHINESE_TRADITIONAL = 31748
Chinese Traditional or Mandarin.

bsLang_SERBIAN_NEUTRAL = 31770
Serbian Neutral.

BSLanguageIDConstants enumeration

Languages which has built-in support currently in the component.

Note: This enum is not provided by the component. You will need to declare the constants in your code to use them.

bsLang_AUTO_SELECT = 0
The language will be selected automatically according to the user locale.

bsLang_ARABIC = 1
Arabic.

bsLang_CHINESE_SIMPLIFIED = 4
Chinese simplified.

bsLang_GERMAN = 7
German.

bsLang_GREEK = 8
Greek.

bsLang_ENGLISH = 9
English.

bsLang_SPANISH = 10
Spanish.

bsLang_FRENCH = 12
French.

bsLang_HEBREW = 13
Hebrew.

bsLang_ITALIAN = 16
Italian.

bsLang_PORTUGUESE = 22
Portuguese.

BSMarginUnitsConstants enumeration

Constants used by **MarginUnits** property,

vbInches = 5
  Property values are in inches.

vbMillimeters = 6
  Property values are in millimeters.

BSMarginUnitsForUserConstants enumeration

Constants used by **MarginUnitsForUser** property,

bsMUInches = 5
　　The user will see the margin units in inches.

bsMUMillimeters = 6
　　The user will see the margin units in millimeters.

bsMUUserLocale = 0
　　The user will see the margin units in inches or millimeters, depending on Windows locale settings.

BSMenuButtonIconConstants enumeration

Constants for **MenuButtonIcon** property.

bsIconStyleDownArrow = 2
   The button will show a down arrow similar to that of a ComboBox control.

bsIconStyleEllipsis = 3
   The button will show three horizontally arranged dots.

bsIconStyleHamburger = 0
   The button will show three horizontal lines, one above the other.

bsIconStyleVerticalEllipsis = 1
   The button will show three vertically arranged dots.

BSMultilineTextAlignmentConstants enumeration

Constants to set the alignment of multiline texts on headers and footers.

bsMTAlignAccordingToPosition = -1
    The text alignment is according to the **Position** argument.

bsMTAlignCenter = 2
    Text is centered.

bsMTAlignLeft = 0
    Text aligned to the left.

bsMTAlignRight = 1
    Text aligned to the right.

BSOpCodeConstants enumeration

Constants to be used in the PaintPicture method of the Printer object for the **Opcode** parameter.

It includes the original raster op constants and adds some new options.

bsSMHalfTone = 0
> Stretch mode: HalfTone, what means better quality for stretching.
>
> It is the default setting, and was not available in the Original VB6's Printer object. To set this object to work as the original (lower quality when stretching images), set it to **bsSMVB6Default**.

vbNotSrcErase = 1114278
> Original from VB6.
> Raster operation: Inverts the result of combining the destination and source bitmaps by using OR.

vbNotSrcCopy = 3342344
> Original from VB6.
> Raster operation: Copies the inverted source bitmap to the destination.

vbSrcErase = 4457256
> Original from VB6.
> Raster operation: Inverts the destination bitmap and combines the result with the source bitmap by using AND.

vbDstInvert = 5570569
> Original from VB6.
> Raster operation: Inverts the destination bitmap.

vbPatInvert = 5898313
> Original from VB6.
> Raster operation: Combines the destination bitmap with the pattern by using XOR.

vbSrcInvert = 6684742
> Original from VB6.
> Raster operation: Combines pixels of the destination and source bitmaps by using XOR.

vbSrcAnd = 8913094
> Original from VB6.
> Raster operation: Combines pixels of the destination and source bitmaps by using AND.

vbMergePaint = 12255782
> Original from VB6.
> Raster operation: Combines the inverted source bitmap with the destination bitmap by using OR.

vbMergeCopy = 12583114

Original from VB6.
Raster operation: Combines the pattern and the source bitmap.

vbSrcCopy = 13369376
Original from VB6.
Raster operation: Copies the source bitmap to the destination bitmap.

vbSrcPaint = 15597702
Original from VB6.
Raster operation: Combines pixels of the destination and source bitmaps by using OR.

vbPatCopy = 15728673
Original from VB6.
Raster operation: Copies the pattern to the destination bitmap.

vbPatPaint = 16452105
Original from VB6.
Raster operation: Combines the inverted source bitmap with the pattern by using OR. Combines the result of this operation with the destination bitmap by using OR.

bsSMBlackOnWhite = 16777216
Stretch mode: Performs a Boolean AND operation using the color values for the eliminated and existing pixels. If the bitmap is a monochrome bitmap, this mode preserves black pixels at the expense of white pixels.

bsSMVB6Default = 16777216
Stretch mode: the same as **bsSMBlackOnWhite**.

bsSMWhiteOnBlack = 33554432
Stretch mode: Performs a Boolean OR operation using the color values for the eliminated and existing pixels. If the bitmap is a monochrome bitmap, this mode preserves white pixels at the expense of black pixels.

bsSMColorOnColor = 50331648
Stretch mode: Deletes the pixels. This mode deletes all eliminated lines of pixels without trying to preserve their information.

bsSMQualityStretch = 83886080
Stretch mode: Uses an enhanced stretching routine (Bicubic) that produces best quality, at expense of processing time.

bsTransparentColor = 134217728
Stretch mode: the image is painted with a transparent background color. The color used as transparent will be the first pixel of the image, the one at the location (0,0), the first pixel of left-top corner.

BSPageNumbersPositionConstants enumeration

Constants to set the position of the page numbers.

bsPosBottomCenter = 3
>   Page numbers are printed horizontally centered at the bottom.

bsPosBottomInside = 5
>   Page numbers are printed at the bottom, odd-numbered pages on the left, and even-numbered pages on the right.

bsPosBottomLeft = 2
>   Page numbers are printed on the left side at the bottom.

bsPosBottomOutside = 4
>   Page numbers are printed at the bottom, odd-numbered pages on the right, and even-numbered pages on the left.

bsPosBottomRight = 1
>   Page numbers are printed on the right side at the bottom.

bsPosDefault = 0
>   Page numbers are printed on the right side at the bottom unless there is a footer set in that position, in which case they are printed on the left side.

bsPosTopCenter = 8
>   Page numbers are printed horizontally centered at the top.

bsPosTopInside = 10
>   Page numbers are printed at the top, odd-numbered pages on the left, and even-numbered pages on the right.

bsPosTopLeft = 7
>   Page numbers are printed on the left side at the Top.

bsPosTopOutside = 9
>   Page numbers are printed at the top, odd-numbered pages on the right, and even-numbered pages on the left.

bsPosTopRight = 6
>   Page numbers are printed on the right side at the Top.

BSPageNumbersPositionFlagsConstants enumeration

Constants to set flags that determine which position options the user will be able to select or not for page numbers.

bsPFAllAvailable = -1
All the flags together.

bsPFAllBottomAvailable = 31
All bottom flags combined.

bsPFAllTopAvailable = 992
All top flags combined.

bsPFBottomCenterAvailable = 4
The user will have the option to select to print the page numbers at the bottom center position. (There may be other factors that could make this option unavailable).

bsPFBottomInsideAvailable = 16
The user will have the option to select to print the page numbers at the bottom inside position. (There may be other factors that could make this option unavailable).

bsPFBottomLeftAvailable = 2
The user will have the option to select to print the page numbers at the bottom left position. (There may be other factors that could make this option unavailable).

bsPFBottomOutsideAvailable = 8
The user will have the option to select to print the page numbers at the bottom outside position. (There may be other factors that could make this option unavailable).

bsPFBottomRightAvailable = 1
The user will have the option to select to print the page numbers at the bottom right position. (There may be other factors that could make this option unavailable).

bsPFHidePageNumbersAvailable = 65536
The user will have the option to hide the page numbers. (There may be other factors that could make this option unavailable).

bsPFTopCenterAvailable = 128
The user will have the option to select to print the page numbers at the top center position. (There may be other factors that could make this option unavailable).

bsPFTopInsideAvailable = 512
The user will have the option to select to print the page numbers at the top inside position. (There may be other factors that could make this option unavailable).

bsPFTopLeftAvailable = 64
The user will have the option to select to print the page numbers at the top left position. (There may be other factors that could make this option unavailable).

bsPFTopOutsideAvailable = 256

The user will have the option to select to print the page numbers at the top outside position. (There may be other factors that could make this option unavailable).

bsPFTopRightAvailable = 32

The user will have the option to select to print the page numbers at the top right position. (There may be other factors that could make this option unavailable).

BSPageViewConstants enumeration

Constants used by the **PageView** property.

bsPVFullHeight = 3
The page is adjusted to the available height. If the width is enough, it can show two or more pages.

bsPVFullWidth = 2
The page is adjusted to the available width.

bsPVNormalSize = 1
The size in which the page is displayed is the one that corresponds to the actual size of the page.

bsPVSixPages = 6
It displays up to six pages.

bsPVThreePages = 5
Three pages are displayed.

bsPVTwelvePages = 7
It displays up to twelve pages.

bsPVTwoPages = 4
Two pages are displayed.

BSPositionPropertyConstants enumeration

Constants to select the position that is referred.

bsLeft = 0
  Used to get the Left position of an element.

bsTop = 1
  Used to get the Top position of an element.

bsWidth = 2
  Used to get the Width of an element.

bsHeight = 3
  Used to get the Height of an element.

BSPreviewPagesAccuracyConstants enumeration

Constants used to set the **PreviewPagesAccuracy** property.

bsPAAccurateDrawing = 1
> The drawing of the pages in the **Print Preview dialog** box is slower but shows what will be printed without any scaling mismatch.

bsPAFastDrawing = 0
> The drawing of the pages in the **Print Preview dialog** box is faster but can be some minor issues with the text scaling of long texts while previewing the pages.

BSPrintCellsFormattingConstants enumeration

Constants used in the **PrintGrid** method for the **PrintCellsFormatting** parameter.

bsPCFIgnoreAll = 31
    Ignore all cell formatting.

bsPCFIgnoreBackColor = 2
    Ignore background colors of cells set individually (**CellBackColor**).

bsPCFIgnoreFonts = 4
    Ignore fonts of cells set individually (**CellFont**).

bsPCFIgnoreForeColor = 1
    Ignore foreground colors of cells set individually (**CellForeColor**).

bsPCFIgnorePictures = 16
    Ignore pictures of cells (**CellPicture**).

bsPCFIgnoreTextAlignment = 8
    Ignore text alignment of cells set individually (**CellAlignment**).

bsPCFPrintAll = 0
    Print all cell formatting.

BSPrinterFlagsConstants enumeration

Constants used by the **Flags** property.

Read full info: **https://docs.microsoft.com/en-us/windows/win32/api/commdlg/ns-commdlg-printdlgexa**

bsPDAllPages = 0
> The default flag that indicates that the **All** (pages) radio button is initially selected.
>
> This flag is used as a placeholder to indicate that the **bsPDPageNums**, **bsPDSelection**, and **bsPDCurrentPage** flags are not specified.

bsPDCurrentPage = 4194304
> If this flag is set, the **Current Page** radio button is selected instead of **All**.
>
> When the user selects the **Current Page** option, he will print the current page in the Print Preview dialog box. If more than one page is displayed, only the first will be printed.
> If the Print Preview dialog box is not displayed, this option will be disabled.
>
> See also the flag **bsPDNoCurrentPage**

bsPDDisablePrintToFile = 524288
> Disables the **Print to File** check box.

bsPDNoCurrentPage = 8388608
> Disables the **Current Page** radio button.

bsPDNoPageNums = 8
> Disables the **Pages** radio button and the associated edit controls.

bsPDNoSelection = 4
> Disables the **Selection** radio button.

bsPDNoWarning = 128
> Prevents the warning message from being displayed when an error occurs.

bsPDPageNums = 2
> If this flag is set, the **Pages** radio button is selected. If none of the **bsPDPageNums**, **bsPDSelection**, or **bsPDCurrentPage** flags is set, the **All** radio button is selected.
>
> The user can specify pages or page ranges, for example:
> 1,4,7
> 1,3,5,10-12
> 3,7-9,13-17
>
> (without spaces)

bsPDPrintToFile = 32
> If this flag is set, the **Print to File** check box is selected.

See also the flag **bsPDShowPrintToFile**

bsPDSelection = 1

If this flag is set, the **Selection** radio button is selected instead of **All**.

When the user selects **Selection** option, he will print the pages that are currently displayed in the Print Preview dialog box.
If the Print Preview dialog box is not displayed, this option will be disabled.

See also the flag **bsPDNoSelection**

bsPDShowPrintToFile = 256

Shows the **Print to File** check box.

Note: this option is added and is not available in the original PrintDlgEx dialog (API), it instead has the HidePrintToFile flag. But here that check box was set hidden by default, and hence the flags are changed.

BSPrintExOptionsConstants enumeration

Options for the **PrintEx** method intended to emulate the ones of the **Printer.Print** method **;** and **,** (semicolon, and comma).

bsAddTab = 2

A tab will be added after the text. Similar to adding a comma after the **Printer.Print** method.

bsLineFeed = 0

A line feed will be added after the text.

bsNoLineFeed = 1

No line feed will be added after the text. Similar to adding a semicolon after the **Printer.Print** method.

BSRememberUserPreferencesConstants enumeration

Constants used by the **RememberUserPreferences** property.

bsSPNever = 0

It doesn't save the settings that the user sets. The next time they won't be remembered.

bsSPWhenPrinted = 1

Saves the settings that the user has set, but only if the report was printed. If the user changed something but finally did not print, they are not saved.
If saved, the next time the report will be configured with these settings.

bsSPAlways = 2

Saves the settings that the user has set on the dialogs, regardless the report was actually printed or not.
The next time the report will be configured with these settings.

BSShowPrinterDialogResultConstants enumeration

Constants used for the return value of **ShowPrinter** dialog box.

bsPDResultApply = 2
> The user pressed the Apply button in the Printer dialog box.

bsPDResultCancel = 0
> The user pressed the Cancel button in the Printer dialog box.

bsPDResultPrint = 1
> The user pressed the Print button in the Printer dialog box.

BSShowSketchConstants enumeration

Constants used by the **ShowSketch** property.

bsSKAlways = 1
It will always show the sketch, regardless of the page count or time elapsed.

bsSKNever = 0
It will never show the sketch.

bsSKWithLargeDocument = 2
It will show the sketch, in the first drawn if the report has 14 pages or more, and in any further redrawn if the last time it took 1 second or more to draw the report.

BSToolbarIconIdentifierConstants enumeration

Constants that are used to identify the icons of the buttons on the top toolbar of the **Print Preview dialog box**.

bsTBIAuxiliary = 3
Images for the "Auxiliary" button.

bsTBIDecreaseScale = 12
Images for the "DecreaseScale" button.

bsTBIDecreaseScaleAlt = 13
Alternative images for the "DecreaseScale" button. See **UseAltScaleIcons** property.

bsTBIFirstPage = 16
Images for the "FirstPage" button.

bsTBIFormat = 5
Images for the "Format" button.

bsTBIIncreaseScale = 14
Images for the "IncreaseScale" button.

bsTBIIncreaseScaleAlt = 15
Alternative images for the "IncreaseScale" button. See **UseAltScaleIcons** property.

bsTBILastPage = 19
Images for the "LastPage" button.

bsTBIMargins = 2
Images for the "Margins" button.

bsTBINextPage = 18
Images for the "NextPage" button.

bsTBIOrientationLandscape = 7
Images for the "OrientationLandscape" button.

bsTBIOrientationPortrait = 6
Images for the "OrientationPortrait" button.

bsTBIPageNumbers = 4
Images for the "PageNumbers" button.

bsTBIPageSetup = 1
Images for the "PageSetup" button.

bsTBIPreviousPage = 17
Images for the "PreviousPage" button.

bsTBIPrint = 0
Images for the "Print" button.

bsTBIViewFullHeight = 10

bsTBIViewFullWidth = 9

bsTBIViewMultiplePages = 11
Images for the "ViewMultiplePages" button.

bsTBIViewNormalSize = 8
Images for the "ViewNormalSize" button.

BSToolbarIconsSizeConstants enumeration

Constants used by the **ToolbarIconsSize** property

bsPPTIconsAuto = 4
>The size of the icons is selected automatically (small, medium or big) according to the screen size.

bsPPTIconsAutoAndUserCanChange = 0
>The size of the icons is selected automatically (small, medium or big) according to the screen size, and the user has the option to change it by selecting a menu option. The menu is displayed by right clicking on the top toolbar or by clicking the Menu button (see **MenuButtonVisible** property).

bsPPTIconsBig = 7
>The size of the icons is big, and the user cannot change it.

bsPPTIconsBigAndUserCanChange = 3
>The default size of the icons is big, and the user has the option to change it by selecting a menu option. The menu is displayed by right clicking on the top toolbar or by clicking the Menu button (see **MenuButtonVisible** property).

bsPPTIconsMedium = 6
>The size of the icons is medium, and the user cannot change it.

bsPPTIconsMediumAndUserCanChange = 2
>The default size of the icons is medium, and the user has the option to change it by selecting a menu option. The menu is displayed by right clicking on the top toolbar or by clicking the Menu button (see **MenuButtonVisible** property).

bsPPTIconsSmall = 5
>The size of the icons is small, and the user cannot change it.

bsPPTIconsSmallAndUserCanChange = 1
>The default size of the icons is small, and the user has the option to change it by selecting a menu option. The menu is displayed by right clicking on the top toolbar or by clicking the Menu button (see **MenuButtonVisible** property).

BSUserInterfaceTextsIDConstants enumeration

Identifiers for the caption texts and strings used in the component to show to the final user.
In each constant comment is the default English caption.

When translating to a new language, many of them are standard captions that can be found with **https://www.microsoft.com/en-us/language/search**.

Note: This enum is not provided by the component. You will need to declare the constants in your code to use them.

bsPPUI_General_CloseButton_Caption = 2000
> English text: "Close"

> It is a generic caption for the 'Close' button.

bsPPUI_General_OKButton_Caption = 2100
> English text: "OK"

> It is a generic caption for the 'OK' button.

bsPPUI_General_CancelButton_Caption = 2200
> English text: "Cancel"

> It is a generic caption for the 'Cancel' button.

bsPPUI_General_Unnamed_FileName = 2300
> English text: "Unnamed"

> Default file name when the user did not supply a name.

bsPPUI_General_PageNumbersFormatStrings_Count = 4000
> English text: "21"

> This must be the number of format strings for the page numbers that will be available. It must be a numeric value.
> The format strings will be fetched with ID
> **bsPPUI_General_PageNumbersFormatString_Item** to N -1, where N is the number specified in **bsPPUI_General_PageNumbersFormatStrings_Count**.

bsPPUI_General_PageNumbersFormatString_Item = 4100
> English text: "{#}"

> Base ID for the page numbers format strings. The format strings start with this ID the first one and go to N -1 for the last one, where N is the number supplied for TextID **bsPPUI_General_PageNumbersFormatStrings_Count**.

> To see the different format string for English, read the description of the **PageNumbersFormat** property.

bsPPUI_General_PrintDocumentTooLongWarning_PageText = 4400
English text: "Document too long to print complete, done to the page {####}."

This text will be printed in a page added at the end of the report when attempting to print more than 2000 pages.

{####} is a placeholder, do not modify it.

bsPPUI_General_PrintDocumentTooLongWarning_MsgBoxWarning = 4500
English text: "Document too long, will not be printed completely."

This message appears to the user when attempting to print more than 2000 pages in a report.

bsPPUI_General_PrinterNotAvailable1 = 4600
English text: "The printer '{Printer_Name}' is not available at this time, would you like to print to the default printer?"

This message will be displayed when the user attempts to print to a printer device that is no longer available.

{Printer_Name} is a placeholder for the printer device name, do not modify it.

bsPPUI_General_PrinterNotAvailable2 = 4700
English text: "The printer '{Printer_Name}' is not available at this time, cannot print."

This message will be displayed when the user attempts to print to a printer device that is no longer available and no other printer is available or installed.

{Printer_Name} is a placeholder for the printer device name, do not modify it.

bsPPUI_General_PageNumbers_Dont_Exists_Message = 4800
English text: "The specified page number(s) do not exist in the document."

This message will be displayed when the user on the **Print dialog** specifies one or more page numbers that do not exist in the report.

bsPPUI_General_SaveToFile_PrinterNotInstalled = 4900
English text: "The '{Printer_Name}' printer driver is not installed on this computer. Cannot save the file."

This message will be displayed to the user when, with the **SaveToFile** method, the printer specified by the **DeviceNameToUse** parameter is not installed on the computer.

{Printer_Name} is a placeholder for the printer device name, do not modify it.

bsPPUI_General_GridReportStyleNames_Basic = 6000
English text: "Basic"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Basic_Gray = 6100
English text: "Basic Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Basic_Red = 6200
English text: "Basic Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Basic_Green = 6300
English text: "Basic Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Basic_Yellow = 6400
English text: "Basic Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Modern = 8000
English text: "Modern"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Modern_Gray = 8100
English text: "Modern Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Modern_Red = 8200
English text: "Modern Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Modern_Green = 8300
English text: "Modern Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Modern_Yellow = 8400
English text: "Modern Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Grid = 10000
English text: "Grid"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Grid_Gray = 10100
English text: "Grid Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Grid_Red = 10200
English text: "Grid Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Grid_Green = 10300
English text: "Grid Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Grid_Yellow = 10400
English text: "Grid Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Closed = 12000
English text: "Closed"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Closed_Gray = 12100
English text: "Closed Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Closed_Red = 12200
English text: "Closed Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Closed_Green = 12300
English text: "Closed Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Closed_Yellow = 12400
English text: "Closed Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Horizontal = 14000
English text: "Horizontal"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Horizontal_Gray = 14100
English text: "Horizontal Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Horizontal_Red = 14200
English text: "Horizontal Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Horizontal_Green = 14300
English text: "Horizontal Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Horizontal_Yellow = 14400
English text: "Horizontal Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Free = 16000
English text: "Free"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Free_Gray = 16100
English text: "Free Grey"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Free_Red = 16200
English text: "Free Red"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Free_Green = 16300
English text: "Free Green"

Name for this style shown in the FormatOptions dialog.

bsPPUI_General_GridReportStyleNames_Free_Yellow = 16400
English text: "Free Yellow"

Name for this style shown in the FormatOptions dialog.

bsPPUI_PreviewDialog_Caption = 18000
English text: "Print preview"

It is the title bar caption that the **Print Preview dialog box** will display when the **PreviewDialogCaption** property is not set.

bsPPUI_PreviewDialog_mnuView2p_Caption = 18100
English text: "View 2 pages"

Is the text of the menu item to display two pages.

bsPPUI_PreviewDialog_mnuView3p_Caption = 18200
English text: "View 3 pages"

Is the text of the menu item to display three pages.

bsPPUI_PreviewDialog_mnuView6p_Caption = 18300
English text: "View 6 pages"

Is the text of the menu item to display six pages.

bsPPUI_PreviewDialog_mnuView12p_Caption = 18400
English text: "View 12 pages"

Is the text of the menu item to display twelve pages.

bsPPUI_PreviewDialog_mnuTopIconsAuto_Caption = 18500
English text: "Automatic icon size"

Is the text of the menu item to set the icon size to automatic.

bsPPUI_PreviewDialog_mnuTopIconsSmall_Caption = 18600
English text: "Small icons"

Is the text of the menu item to set the icon size to small.

bsPPUI_PreviewDialog_mnuTopIconsMedium_Caption = 18700
English text: "Medium icons"

Is the text of the menu item to set the icon size to medium.

bsPPUI_PreviewDialog_mnuTopIconsLarge_Caption = 18800
English text: "Large icons"

Is the text of the menu item to set the icon size to large.

bsPPUI_PreviewDialog_Icons_Size_Small = 18900
English text: "small"

Is the text that will appear in brackets after the text of
bsPPUI_PreviewDialog_mnuTopIconsAuto_Caption when the aotomatic size of
icons is set to small.

bsPPUI_PreviewDialog_Icons_Size_Medium = 19000
English text: "medium"

Is the text that will appear in brackets after the text of
bsPPUI_PreviewDialog_mnuTopIconsAuto_Caption when the aotomatic size of
icons is set to medium.

bsPPUI_PreviewDialog_Icons_Size_Large = 19100
English text: "large"

Is the text that will appear in brackets after the text of
bsPPUI_PreviewDialog_mnuTopIconsAuto_Caption when the aotomatic size of
icons is set to large.

bsPPUI_PreviewDialog_mnuTopShowBottomToolbar_Caption = 19200
English text: "Show bottom bar"

Is the text of the menu item to set whether to show a bottom bar for the page
navigation buttons or not.

bsPPUI_PreviewDialog_mnuTopShowMargins_Caption = 19300
English text: "Show margins"

Is the text of the menu item to set whether to show the margins on the pages or not.

bsPPUI_PreviewDialog_CurrentlySelected = 19400
English text: "selected"

Is a text used in brackets to identify, in some circumstances, that an item is selected.

375

bsPPUI_PreviewDialog_CurrentlyShown = 19500
English text: "currently shown"

Is a text used in brackets to identify, in some circumstances, that an element is currently shown.

bsPPUI_PreviewDialog_lblPageOrientation_Caption = 19600
English text: "Page orientation:"

The text of the label for 'Page orientation:' at the toolbar.

bsPPUI_PreviewDialog_lblView_Caption = 19700
English text: "View:"

The text of the label for 'View:' at the toolbar.

bsPPUI_PreviewDialog_lblScale_Caption = 19800
English text: "Scale:"

The text of the label for 'Scale:' at the toolbar.

bsPPUI_PreviewDialog_CurrentPage_Singular_Caption = 19900
English text: "Page: {#} of {##}"

Template used when displaying one page, that is shown at the toolbar (on the bottom or top toolbar, depending on the setting of **ShowBottomToolbar** property and user setting).

{#} is the placeholder for the current page number that is on the screen, and {##} for the total page count.

bsPPUI_PreviewDialog_CurrentPage_Plural_Caption = 20000
English text: "Pages: {#} of {##}"

Template used when displaying more than one page, that is shown at the toolbar (on the bottom or top toolbar, depending on the setting of **ShowBottomToolbar** property and user setting).

{#} is the placeholder for the current pages numbers that are on the screen, and {##} for the total page count.

bsPPUI_PreviewDialog_PreparingDoc_Caption_LoadingPreview = 20200
English text: "Loading preview..."

It is the status text at the bottom in the **Print Preview dialog box** that is displayed while the report is prepared.

bsPPUI_PreviewDialog_PreparingDoc_Caption_PreparingDocument = 20300
English text: "Preparing document..."

It is an aternative status text that appear at the bottom in the **Print Preview dialog box** in some circumstances while the report is prepared.

bsPPUI_PreviewDialog_cmdPrintInDefaultPrinter_Caption = 20400

English text: "Print to {Printer_Name}"

Caption of the button that appears at the toolbar when the **QuickPrintButtonVisible** property is set to **True** and there is enough space to show it.

{Printer_Name} is a placeholder for the printer device name, do not modify it.

bsPPUI_PreviewDialog_cmdClose_Caption = 20500
English text: "Close print preview"

Caption of the button that appears at the toolbar to close the **Print pPreview dialog**.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_Print = 20600
English text: "Print"

ToolTip text of the toolbar button to print.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_PageSetup = 20700
English text: "Page setup"

ToolTip text of the toolbar button to show the **Page setup dialog**.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_Auxiliary = 20800
English text: ""

ToolTip text of the toolbar "Auxiliary" button that is shown when **AuxiliaryButtonVisible** property is set to **True**.

There is a property to set the ToolTip text of this button also: **AuxiliaryButtonToolTipText**. The rule is that if the **AuxiliaryButtonToolTipText** property is set to "", then this localized text got from the **GetLocalizedText** event is used, otherwise it is used the text from the property.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_PageNumbersOptions = 20900
English text: "Page numbers options"

ToolTip text of the toolbar button to show the **PageNumbersOptions dialog**.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_Format = 21000
English text: "Format"

ToolTip text of the toolbar button to show the **Format dialog**.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_OrientationPortrait = 21100
English text: "Portrait orientation"

ToolTip text of the toolbar button to set the orientation to portrait.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_OrientationLandscape = 21200
English text: "Landscape orientation"

ToolTip text of the toolbar button to set the orientation to landscape.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_OrientationCantBeChanged = 21300
English text: "The page orientation cannot be changed"

ToolTip text that will be used on the toolbar buttons that are intended to change the page orientation when the device does not support page orientation change.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_ViewNormalSize = 21400
English text: "View normal page size"

ToolTip text of the optional toolbar button to show the page at its normal size.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_ViewFullWidth = 21500
English text: "View full width"

ToolTip text of the toolbar button to show the page(s) at full width.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_ViewFullHeight = 21600
English text: "View full height"

ToolTip text of the toolbar button to show the page(s) at full height.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_ViewMultiplePages = 21700
English text: "Show multiple pages"

ToolTip text of the toolbar button to show multiple pages.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_DecreaseScale = 21800
English text: "Decrease fonts and elements size"

Tooltip text for the toolbar button to scale down.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_IncreaseScale = 21900
English text: "Increase fonts and elements size"

Tooltip text for the toolbar button to scale up.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_ShowMenu = 22000
English text: "Open menu"

ToolTip text of the toolbar button to show the menu.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_FirstPage = 22100
English text: "First page"

ToolTip text of the toolbar button to go to the fist page.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_PreviousPage_Singular = 22200
English text: "Previous page"

ToolTip text of the toolbar button to go to the previous page used when showing one page.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_PreviousPage_Plural = 22300
English text: "Previous pages"

ToolTip text of the toolbar button to go to the previous page used when showing more than one page.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_NextPage_Singular = 22400
English text: "Next page"

ToolTip text of the toolbar button to go to the next page used when showing one page.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_NextPage_Plural = 22500
English text: "Next pages"

ToolTip text of the toolbar button to go to the next page used when showing more than one page.

bsPPUI_PreviewDialog_ToolTipText_ToolbarButton_LastPage = 22600
English text: "Last page"

ToolTip text of the toolbar button to go to the last page.

bsPPUI_PreviewDialog_NonPrintableAreaText = 22700
English text: "Nonprinting region"

Is the text used to show to the user, when the **Print preview dialog** is showing margins, the non-printable regions on the page (if there is any).

bsPPUI_PreviewDialog_Inches_Abbreviation = 22800
English text: """

Abbreviation for inches, used in the tooltip to show the size of the margins.

bsPPUI_PreviewDialog_Millimeters_Abbreviation = 22900
English text: "mm"

Abbreviation for millimeters, used in the tooltip to show the size of the margins.

bsPPUI_ProgressWindow_lblMessage_Caption_Start_PrintPreview = 24000
English text: "Loading preview"

Caption used in the progress window.

bsPPUI_ProgressWindow_lblMessage_Caption_Progress_PrintPreview = 24100
English text: "Loading preview, page {#}"

Caption used in the progress window.

{#} is a placeholder for the page current number being processed.

bsPPUI_ProgressWindow_lblMessage_Caption_Start_General = 24200
English text: "Generating document"

Caption used in the progress window.

bsPPUI_ProgressWindow_lblMessage_Caption_Progress_General = 24300
English text: "Generating document, page {#}"

Caption used in the progress window.

{#} is a placeholder for the page current number being processed.

bsPPUI_ProgressWindow_MsgBox_Cancel_Preview = 24400
English text: "Cancel the operation?"

Text of the message asking the user whether to cancel the current operation.

bsPPUI_FormatOptionsDialog_Caption = 26000
English text: "Print format"

Text of the caption of the **Format dialog** box.

bsPPUI_FormatOptionsDialog_TabCaption_1 = 26100
English text: "Options"

Caption of the first tab of the dialog.

bsPPUI_FormatOptionsDialog_TabCaption_2 = 26200
English text: "Style"

Caption of the second tab of the dialog.

bsPPUI_FormatOptionsDialog_lblScale_Caption = 26300
English text: "Scale:"

Caption of the Label for Scale.

bsPPUI_FormatOptionsDialog_lblColorMode_Caption = 26400
English text: "Color mode:"

Caption of the Label for Color mode.

bsPPUI_FormatOptionsDialog_lblPageNumbersPosition_Caption = 26500
English text: "Page numbers position:"

Caption of the Label for Page numbers position.

bsPPUI_FormatOptionsDialog_lblPageNumberFormat_Caption = 26600
English text: "Page number format:"

Caption of the Label for Page number format.

bsPPUI_FormatOptionsDialog_lblPageNumbersFont_Caption = 26700
English text: "Page numbers font:"

Caption of the Label for Page numbers font.

bsPPUI_FormatOptionsDialog_lblGridPosition_Caption = 26800
English text: "Data grid position:"

Caption of the Label for Data grid position.

bsPPUI_FormatOptionsDialog_cboColorMode_PrinterDefault = 26900
English text: "Printer default"

In the Color mode combo box, the option for "Printer default".

bsPPUI_FormatOptionsDialog_cboColorMode_GrayScale = 27000

English text: "Grayscale"

In the Color mode combo box, the option for "Grayscale".

bsPPUI_FormatOptionsDialog_cboColorMode_Color = 27100
English text: "Color"

In the Color mode combo box, the option for "Color".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_BottomRight = 27200
English text: "Bottom right"

In the Page numbers position combo box, the option for "Bottom right".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_BottomLeft = 27300
English text: "Bottom left"

In the Page numbers position combo box, the option for "Bottom left".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_BottomCenter = 27400
English text: "Bottom center"

In the Page numbers position combo box, the option for "Bottom center".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_BottomOutside = 27500
English text: "Bottom outside"

In the Page numbers position combo box, the option for "Bottom outside". See
BSPageNumbersPositionConstants.

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_BottomInside = 27600
English text: "Bottom inside"

In the Page numbers position combo box, the option for "Bottom inside". See
BSPageNumbersPositionConstants.

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_TopRight = 27700
English text: "Top right"

In the Page numbers position combo box, the option for "Top right".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_TopLeft = 27800
English text: "Top left"

In the Page numbers position combo box, the option for "Top left".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_TopCenter = 27900
English text: "Top center"

In the Page numbers position combo box, the option for "Top center".

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_TopOutside = 28000
English text: "Top outside"

In the Page numbers position combo box, the option for "Top outside". See
BSPageNumbersPositionConstants.

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_TopInside = 28100
English text: "Top inside"

In the Page numbers position combo box, the option for "Top inside". See
BSPageNumbersPositionConstants.

bsPPUI_FormatOptionsDialog_cboPageNumbersPosition_HidePageNumbers = 28200
English text: "Hide page numbers"

In the Page numbers position combo box, the option for "Hide page numbers" (not to
print the page numbers).

bsPPUI_FormatOptionsDialog_fpcPageNumbers_ButtonToolTipTextDefault = 28300
English text: "Choose font"

ToolTip text of the button to select a different font.

bsPPUI_FormatOptionsDialog_fpcPageNumbers_DrawSample_Bold = 28400
English text: "bold"

The ToolTip text of the font sample shows the font name and its attributes. This is the
text for "bold".

bsPPUI_FormatOptionsDialog_fpcPageNumbers_DrawSample_Italic = 28500
English text: "italic"

The ToolTip text of the font sample shows the font name and its attributes. This is the
text for "italic".

bsPPUI_FormatOptionsDialog_cboGridPosition_Centered = 28600
English text: "Centered"

In the Data grid position combo box, the option for "Centered".

bsPPUI_FormatOptionsDialog_cboGridPosition_Left = 28700
English text: "Left"

In the Data grid position combo box, the option for "Left".

bsPPUI_FormatOptionsDialog_cboGridPosition_Right = 28800
English text: "Right"

In the Data grid position combo box, the option for "Right".

bsPPUI_FormatOptionsDialog_cboGridPosition_Stretch = 28900
English text: "Stretch"

In the Data grid position combo box, the option for "Stretch" (stretch the grid to fill the
width of the page).

bsPPUI_FormatOptionsDialog_cboGridPosition_EnlargeLastColumn = 29000
English text: "Enlarge the last column"

In the Data grid position combo box, the option for "Enlarge the last column" (to fill
the width of the page).

bsPPUI_FormatOptionsDialog_cboGridPosition_ToolTipText = 29100
English text: "It only has effect when the data grid is narrower than the page"

ToolTip text of the Data grid position combo box.

bsPPUI_FormatOptionsDialog_lblStyle_Caption = 29200
English text: "Style:"

In the tab 2, the caption of the Label for the combo box for selecting the Style (grid style).

bsPPUI_FormatOptionsDialog_cboStyle_List_Style = 29300
English text: "Style"

Text used when a style does not have a name. The name displayed to the user is composed with this text and a number, such as "Style 3".

bsPPUI_FormatOptionsDialog_cboStyle_List_CustomStyle = 29400
English text: "Custom style"

Text used for naming custom styles set by the user. The names are composed with this text and a number, such as "Custom style 2".

bsPPUI_FormatOptionsDialog_cboStyle_List_Customize = 29500
English text: "Customize"

Text that appears at the end of the list of styles, allowing the user to create a new one.

bsPPUI_FormatOptionsDialog_cmdDeleteCustomStyle_ToolTipText = 29600
English text: "Delete style"

Caption of the button to delete the currently selected style. The styles that the user can delete are only the custom ones.

bsPPUI_FormatOptionsDialog_lblLineThickness_Caption = 29700
English text: "Line thickness:"

Caption of the Label for Line thickness.

bsPPUI_FormatOptionsDialog_txtLineThickness_ToolTipText = 29800
English text: "Change line thickness (general)"

ToolTip text for the text box to enter the line thickness, that is applied for all the lines but the header separator.

bsPPUI_FormatOptionsDialog_ValidateLineThickness_Message = 29900
English text: "The line thickness value must be between 1 and 80"

Message text to show the user when an invalid line thickness value was entered. The allowed values are between 1 and 80.

bsPPUI_FormatOptionsDialog_chkDrawHeaderBackground_Caption = 30000
English text: "Headers background"

Caption of the check box to select whether to paint the headers background or not.

bsPPUI_FormatOptionsDialog_chkDrawFixedColsBackground_Caption = 30100
English text: "Fixed columns Backg."

Caption of the check box to select whether to paint the fixed columns background or not.

bsPPUI_FormatOptionsDialog_cmdFixedSectionsBackgroundColor_ToolTipText = 30200
English text: "Change background color of headers (and / or fixed columns)"

ToolTip text of the button to change the background color of headers and/or fixed columns.

bsPPUI_FormatOptionsDialog_chkDrawOtherBackgrounds_Caption = 30300
English text: "Other background colors"

Caption of the check box to select whether to paint the background of other cells (that have colored background in the original grid) or not.

bsPPUI_FormatOptionsDialog_chkDrawOuterBorder_Caption = 30400
English text: "Outer border"

Caption of the check box to select whether to draw the outer border or not.

bsPPUI_FormatOptionsDialog_chkDrawHeaderBorder_Caption = 30500
English text: "Headers borders"

Caption of the check box to select whether to draw the headers border or not.

bsPPUI_FormatOptionsDialog_chkDrawColumnsHeadersLines_Caption = 30600
English text: "Column headers lines"

Caption of the check box to select whether to draw the column lines in the headers or not.

bsPPUI_FormatOptionsDialog_chkDrawColumnsDataLines_Caption = 30700
English text: "Columns data lines"

Caption of the check box to select whether to draw the column lines in the data section or not.

bsPPUI_FormatOptionsDialog_chkDrawRowLines_Caption = 30800
English text: "Row lines"

Caption of the check box to select whether to draw the row lines or not.

bsPPUI_FormatOptionsDialog_chkDrawHeaderSeparatorLine_Caption = 30900
English text: "Headers separator:"

Caption of the check box to select whether to draw a line that separates the header from the data section or not.

bsPPUI_FormatOptionsDialog_txtHeaderSeparatorLineThickness_ToolTipText = 31000
English text: "Headers separator line thickness"

ToolTip text for the text box to enter the header separator line thickness.

bsPPUI_FormatOptionsDialog_VariousChangeColorCommandButtons_ToolTipText = 31100
English text: "Change color"

ToolTip text for all the buttons used to change colors.

bsPPUI_FormatOptionsDialog_ValidateHeaderSeparatorLineThickness_Message = 31200
English text: "The line thickness value must be between 1 and 80"

Message text to show the user when an invalid line thickness value was entered for the header separator. The allowed values are between 1 and 80.

bsPPUI_FormatOptionsDialog_lblSample_Caption = 31300
English text: "Sample:"

Caption of the Label for the sample drawing.

bsPPUI_FormatOptionsDialog_DrawSample_Column = 31400
English text: "Column"

Text for one individual header cell in the sample drawing.

bsPPUI_FormatOptionsDialog_DrawSample_Element = 31500
English text: "Element"

Text for each first cell of a row in the sample drawing.

bsPPUI_FormatOptionsDialog_DrawSample_Data = 31600
English text: "Data"

Text for each data cell in the sample drawing.

bsPPUI_PageNumbersOptionsDialog_Caption = 34000
English text: "Page numbers options"

Text of the caption of the **PageNumbersOptions dialog** box.

BSWatermarkPositionUnitsConstants enumeration

Constants used in the **AddWatermark** method to set with which units a watermark is placed on a page.

bsPUPercentageOfPage = 0

 The watermark position and size are set in percentage of the page (of the full page, without taking into account margins, but taking into account the non-printable area).

bsPUPrinterScaleMode = 1

 The watermark position and size are set in the current ScaleMode units of the Printer object. If HandleMargins is True, the position and size is set inside the margins. Printer.ScaleWidth and Printer.ScaleHeight return the full ranges as a reference.

BSWatermarkZPositionConstants enumeration

Constants used in the **AddWatermark** method to set the ZPosition of a watermark.

bsWPBackground = 0
> The watermark is printed in the background of other texts.

bsWPForeground = 1
> The watermark is printed in the foreground of other texts.