

Class *viewer/convertlengths

This class provides several simple unit of measure conversion methods used by the viewer

Properties

PROPERTY Version\$

Methods

FUNCTION InchesToMilliMeters(Length, decimals)

Convert inches to millimeters

Parameter	Description
-----------	-------------

Length	Length in inches
--------	------------------

decimals	Number of decimal places for returned value
----------	---

Returns converted length in millimeters

FUNCTION InchesToMilliMeters(Length)

Convert inches to millimeters (2 decimal places of precision)

Parameter	Description
-----------	-------------

Length	Length in inches
--------	------------------

Returns converted length in millimeters

FUNCTION MilliMetersToInches(Length, decimals)

Convert millimeters to inches

Parameter	Description
-----------	-------------

Length	Length in millimeters
--------	-----------------------

decimals	Number of decimal places for returned value
----------	---

Returns converted length in inches

FUNCTION MilliMetersToInches(Length)

Convert millimeters to inches (4 decimal places of precision)

Parameter	Description
-----------	-------------

Length	Length in millimeters
--------	-----------------------

Returns converted length in inches

Class *viewer/fontfunctions

Properties

PROPERTY DPI

PROPERTY Version\$

Methods

FUNCTION GetSizesForPage\$(FontName\$, StartSize, Width, Length, Margins\$)

Get comma-separated list of sizes for a font for specified page size in the format 'Size = Columns : Rows'

Parameter	Description
-----------	-------------

FontName\$	The name of the font to be used
StartSize	the start size of the font (default is 40)
Width	The width of the page
Length	The length of the page
Margins	The margins (left:top:right:bottom) for the page

Returns A list of the font sizes (including columns and lines) for the page IE: "-40=80:30,-38=88:32"

FUNCTION GetFontSizeForTextToFitRectangle\$(Text\$, FontName\$, FontSize, FontAttrib\$, FontRotation, Width, Length, DefaultFont\$, DefaultDPI, DefaultLPI, DefaultCPI)

Get the size of the font that is required to display a text value within a rectangle of the specified size

Parameter	Description
-----------	-------------

Text\$	The text value to fit within the rectangle
FontName\$	The name of the font to use
FontSize	The initial font size to be used
FontAttrib\$	The font attributes to be used
FontRotation	The rotation of the font
Width	The width (inches) of the rectangular region
Length	The length (inches) of the rectangular region
DefaultFont\$	The default font
DefaultDPI	The default print resolution (dots per inch)
DefaultLPI	The default lines per inch
DefaultCPI	The default characters per inch

Returns A string with the font information in the format: "Size,WidthInGU,HeightInGU"

FUNCTION CreatePageAsBitmap(Width, Length, Margins\$)

FUNCTION PrintPageAsBitmap(Channel, Data\$)

FUNCTION GetMaximums\$(Channel)

FUNCTION DestroyPageAsBitmap(Channel)

Class *viewer/list.name.value

This object contains the logic to manipulate an adhoc list of name value pairs. Number of Name/Value Pairs is limited only by memory and ProvideX Array Limits. (v6 10 million / 2)

Properties

PROPERTY Version\$

PROPERTY Items

Methods

FUNCTION GetItem(Item%, Name\$, Value\$)

Get a Name/Value pair at specified position in the list

Parameter	Description
Item%	The number of the item to be retrieved
Name\$	The name of the item will be returned in this argument
Value\$	The value of the item will be returned in this argument

FUNCTION GetItem(Name\$, Value\$)

Search for an item that has the specified name

Parameter	Description
Name\$	The name of the item will be found
Value\$	The value of the item will be returned in this argument

Returns item number if found, 0 if not

FUNCTION SetItem(Item%, Name\$, Value\$)

Change a Name/Value pair at specied position in the list

Parameter	Description
%Item	The position of the pair in the list
Name\$	The name to be added
Value\$	The value to associate with the name

returns 0 on success, not 0 for error

FUNCTION SetItem(Name\$, Value\$)

Add a new Name/Value pair into the list

Parameter	Description
Name\$	The name to be added
Value\$	The value to associate with the name

Returns The position in the list of the pair, or 0 if failed to add

Class *viewer/messagelib

This object contains the logic to control Message Library entries.

Constructor

Each time a new message library is needed, this object may be used.

Properties

PROPERTY Version\$

PROPERTY Language\$

The language of the message library file.

PROPERTY File\$

The name of the message library file (without the language extension).

Methods

FUNCTION Language(Language\$)

Set the language to be used for the message file.

Parameter	Description
-----------	-------------

Language\$	The file extension to be used (abbreviated language code).
------------	--

Class *viewer/page

Inherits

[*viewer/paperbin](#), [*viewer/papersize](#)

Properties

PROPERTY Version\$

PROPERTY MarginLeft

PROPERTY MarginRight

PROPERTY MarginTop

PROPERTY MarginBottom

PROPERTY DefaultFont\$

PROPERTY DefaultFontSize

PROPERTY DefaultFontAttributes\$

PROPERTY DefaultFontRotation

PROPERTY CurrentFont\$

PROPERTY CurrentFontSize

PROPERTY CurrentFontAttributes\$

PROPERTY CurrentFontRotation

PROPERTY LPI

PROPERTY CPI

PROPERTY ForegroundColour\$

PROPERTY BackgroundColour\$

PROPERTY Mode\$

PROPERTY Pen\$

PROPERTY Fills

PROPERTY Underline

PROPERTY ReverseVideo

PROPERTY SPCPMode

PROPERTY PlusS

PROPERTY DataType

PROPERTY Data\$

PROPERTY Metric

Methods

No Methods

Properties inherited from : *viewer/paperbin

Version\$, PaperBin, PaperBinName\$

Properties inherited from : *viewer/papersize

Version\$, PaperLength, PaperWidth, PaperName\$, PaperSize, Orientation, Orientation\$, Metric

Methods inherited from : *viewer/papersize

Metric()

Class ***viewer/paperbin**

This object is used to store / track info relating to a Paper Bin (Paper Source)

Known SubClasses

[*viewer/page](#)

Properties

PROPERTY Version\$

PROPERTY PaperBin

PROPERTY PaperBinName\$

Methods

No Methods

Class *viewer/paperbins

This object is used to Validate Paper Bins (Paper Sources) and can return the names of the bins given their number.

Properties

PROPERTY Version\$

PROPERTY DMBIN_UPPER

PROPERTY DMBIN_ONLYONE

PROPERTY DMBIN_LOWER

PROPERTY DMBIN_MIDDLE

PROPERTY DMBIN_MANUAL

PROPERTY DMBIN_ENVELOPE

PROPERTY DMBIN_ENVMANUAL

PROPERTY DMBIN_AUTO

PROPERTY DMBIN_TRACTOR

PROPERTY DMBIN_SMALLFMT

PROPERTY DMBIN_LARGE FMT

PROPERTY DMBIN_LARGE CAPACITY

PROPERTY DMBIN_CASSETTE

PROPERTY DMBIN_FORMSOURCE

PROPERTY DMBIN_LAST

PROPERTY DMBIN_FIRST

PROPERTY DMBIN_USER

Methods

FUNCTION GetPaperBin(BinNumber%)

FUNCTION GetPaperBinProperties(BinNumber%, BinName\$)

FUNCTION GetPaperBinNames(BinNames\$)

Class *viewer/papersize

This object is used to store and track info relating to a paper size

Known SubClasses

[*viewer/page](#)

Properties

PROPERTY Version\$

PROPERTY PaperLength

PROPERTY PaperWidth

PROPERTY PaperName\$

PROPERTY PaperSize

PROPERTY Orientation

PROPERTY Orientation\$

PROPERTY Metric

Methods

FUNCTION Metric(Value%)

Class *viewer/papersizes

Properties

PROPERTY Version\$

PROPERTY DMPAPER_LETTER

PROPERTY DMPAPER_LETTERSMAIL

PROPERTY DMPAPER_TABLOID

PROPERTY DMPAPER_LEDGER

PROPERTY DMPAPER_LEGAL

PROPERTY DMPAPER_STATEMENT

PROPERTY DMPAPER_EXECUTIVE

PROPERTY DMPAPER_A3

PROPERTY DMPAPER_A4

PROPERTY DMPAPER_A4SMALL

PROPERTY DMPAPER_A5

PROPERTY DMPAPER_B4

PROPERTY DMPAPER_B5

PROPERTY DMPAPER_FOLIO

PROPERTY DMPAPER_QUARTO

PROPERTY DMPAPER_10X14

PROPERTY DMPAPER_11X17

PROPERTY DMPAPER_NOTE

PROPERTY DMPAPER_ENV_9

PROPERTY DMPAPER_ENV_10

PROPERTY DMPAPER_ENV_11

PROPERTY DMPAPER_ENV_12

PROPERTY DMPAPER_ENV_14

PROPERTY DMPAPER_CSHEET

PROPERTY DMPAPER_DSHEET

PROPERTY DMPAPER_ESHEET

PROPERTY DMPAPER_ENV_DL

PROPERTY DMPAPER_ENV_C5

PROPERTY DMPAPER_ENV_C3

PROPERTY DMPAPER_ENV_C4

PROPERTY DMPAPER_ENV_C6

PROPERTY DMPAPER_ENV_C65

PROPERTY DMPAPER_ENV_B4

PROPERTY DMPAPER_ENV_B5

PROPERTY DMPAPER_ENV_B6

PROPERTY DMPAPER_ENV_ITALY

PROPERTY DMPAPER_ENV_MONARCH

PROPERTY DMPAPER_ENV_PERSONAL

PROPERTY DMPAPER_FANFOLD_US

PROPERTY DMPAPER_FANFOLD_STD_GERMAN

PROPERTY DMPAPER_FANFOLD_LGL_GERMAN

PROPERTY DMPAPER_ISO_B4

PROPERTY DMPAPER_JAPANESE_POSTCARD

PROPERTY DMPAPER_9X11

PROPERTY DMPAPER_10X11

PROPERTY DMPAPER_15X11

PROPERTY DMPAPER_ENV_INVITE

PROPERTY DMPAPER_RESERVED_48

PROPERTY DMPAPER_RESERVED_49

PROPERTY DMPAPER_LETTER_EXTRA

PROPERTY DMPAPER_LEGAL_EXTRA

PROPERTY DMPAPER_TABLOID_EXTRA

PROPERTY DMPAPER_A4_EXTRA

PROPERTY DMPAPER_LETTER_TRANSVERSE

PROPERTY DMPAPER_A4_TRANSVERSE

PROPERTY DMPAPER_LETTER_EXTRA_TRANSVERSE

PROPERTY DMPAPER_A_PLUS

PROPERTY DMPAPER_B_PLUS

PROPERTY DMPAPER_LETTER_PLUS

PROPERTY DMPAPER_A4_PLUS

PROPERTY DMPAPER_A5_TRANSVERSE

PROPERTY DMPAPER_B5_TRANSVERSE

PROPERTY DMPAPER_A3_EXTRA

PROPERTY DMPAPER_A5_EXTRA

PROPERTY DMPAPER_B5_EXTRA

PROPERTY DMPAPER_A2

PROPERTY DMPAPER_A3_TRANSVERSE

PROPERTY DMPAPER_A3_EXTRA_TRANSVERSE

PROPERTY DMPAPER_DBL_JAPANESE_POSTCARD

PROPERTY DMPAPER_A6

PROPERTY DMPAPER_JENV_KAKU2

PROPERTY DMPAPER_JENV_KAKU3

PROPERTY DMPAPER_JENV_CHOU3

PROPERTY DMPAPER_JENV_CHOU4

PROPERTY DMPAPER_LETTER_ROTATED

PROPERTY DMPAPER_A3_ROTATED

PROPERTY DMPAPER_A4_ROTATED

PROPERTY DMPAPER_A5_ROTATED

PROPERTY DMPAPER_B4_JIS_ROTATED

PROPERTY DMPAPER_B5_JIS_ROTATED

PROPERTY DMPAPER_JAPANESE_POSTCARD_ROTATED

PROPERTY DMPAPER_DBL_JAPANESE_POSTCARD_ROTATED

PROPERTY DMPAPER_A6_ROTATED

PROPERTY DMPAPER_JENV_KAKU2_ROTATED

PROPERTY DMPAPER_JENV_KAKU3_ROTATED

PROPERTY DMPAPER_JENV_CHOU3_ROTATED

PROPERTY DMPAPER_JENV_CHOU4_ROTATED

PROPERTY DMPAPER_B6_JIS

PROPERTY DMPAPER_B6_JIS_ROTATED

PROPERTY DMPAPER_12X11

PROPERTY DMPAPER_JENV_YOU4

PROPERTY DMPAPER_JENV_YOU4_ROTATED

PROPERTY DMPAPER_P16K

PROPERTY DMPAPER_P32K

PROPERTY DMPAPER_P32KBIG

PROPERTY DMPAPER_PENV_1

PROPERTY DMPAPER_PENV_2

PROPERTY DMPAPER_PENV_3

PROPERTY DMPAPER_PENV_4

PROPERTY DMPAPER_PENV_5

PROPERTY DMPAPER_PENV_6

PROPERTY DMPAPER_PENV_7

PROPERTY DMPAPER_PENV_8

PROPERTY DMPAPER_PENV_9

PROPERTY DMPAPER_PENV_10

PROPERTY DMPAPER_P16K_ROTATED

PROPERTY DMPAPER_P32K_ROTATED

PROPERTY DMPAPER_P32KBIG_ROTATED

PROPERTY DMPAPER_PENV_1_ROTATED

PROPERTY DMPAPER_PENV_2_ROTATED

PROPERTY DMPAPER_PENV_3_ROTATED

PROPERTY DMPAPER_PENV_4_ROTATED

PROPERTY DMPAPER_PENV_5_ROTATED

PROPERTY DMPAPER_PENV_6_ROTATED

PROPERTY DMPAPER_PENV_7_ROTATED

PROPERTY DMPAPER_PENV_8_ROTATED

PROPERTY DMPAPER_PENV_9_ROTATED

PROPERTY DMPAPER_PENV_10_ROTATED

PROPERTY DMPAPER_LAST

PROPERTY DMPAPER_USER

Methods

FUNCTION GetPaperSize(PaperNumber%)

FUNCTION GetPaperSize(PaperNumber%, PaperName\$)

FUNCTION GetPaperSize(PaperNumber%, PaperName\$, WidthInches, LengthInches, Widthmm, Lengthmm)

FUNCTION GetPaperSize(PaperNumber%, PaperName\$, WidthInches, LengthInches, Widthmm, Lengthmm, WindowsName\$)

FUNCTION GetPaperSizeNames(PaperNames\$)

Class *viewer/parameters

This object contains the logic to add/modify/delete parameter entries. Used by programs to maintain current and default values for variables based on systemWide and/or Machine and/or User Name. Similar to a registry, but stored in an encrypted keyed file.

Default parameters for an Application and its reference are saved with no user and machine name. The default parameter list is the driving list and the user/machine will always get the default values for a variable if there are no user/machine specific entries, or if the specific variable does not exist in the user/machine specific entries, but does in the default. Application and Reference Names may be NULL (not recommended).

The data types are:

Value	Description
B	Boolean. Always returned as either 0 or 1. Either "" or "0" will be returned as 0; all other values will be returned as 1.
N	Numeric
P	System parameter
S	String
***	Global variables can be used as parameters
***	Arrays cannot be used for parameter values.

Because this program sets and retrieves variable names/values directly from the user's current workspace, variable names within this utility use a _._._ preamble to avoid naming conflicts. You should avoid using variable names in your applications that start with "_._._" or "_._._._"

Constructor

Load/store parameters to a specified file.

Properties

PROPERTY Version\$

Methods

FUNCTION PERFORM GetParameters(Application\$, Reference\$)

Load the parameter values stored for an Application and Reference into the current workspace.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters

FUNCTION PERFORM GetParameters(Application\$, Reference\$, Options\$)

Load the parameters stored for an Application and Reference into the current workspace.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters
Options\$	Options to control associating of the parameters to user, machine, or both.

The allowed values for "Options\$" are here:

Value	Description
""	(same as "USER+MACHINE")
"USER"	The parameters are for the current user.
"MACHINE"	The parameters are for the machine and any user
"USER+MACHINE"	The parameters are for the machine and the current user

FUNCTION PERFORM GetDefaultParameters(Application\$, Reference\$)

Load the default parameter values stored for an Application and Reference into the current workspace.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters

FUNCTION PERFORM SetParameters(Application\$, Reference\$)

Set the parameter values and write them to the parameter file for an Application and Reference.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters

FUNCTION PERFORM SetParameters(Application\$, Reference\$, Options\$)

Set the parameter values and write them to the parameter file for an Application and Reference.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters
Options\$	Options to control associating of the parameters to user, machine, or both.

The allowed values for "Options\$" are here:

Value	Description
""	(same as "USER+MACHINE")
"USER"	The parameters are for the current user.
"MACHINE"	The parameters are for the machine and any user
"USER+MACHINE"	The parameters are for the machine and the current user

FUNCTION PERFORM SetDefaultParameters(Application\$, Reference\$)

Set the default parameter values and write them to the parameter file for an Application and Reference.

Parameter	Description
Application\$	The code used to identify the application
Reference\$	The reference code for this specific application parameters

FUNCTION ExistsApplication(Application\$)

Check to determine if parameters exist for a specified application.

Parameter	Description
Application\$	The code used to identify the application

FUNCTION ExistsReference(Application\$, Reference\$)

Check to determine if parameters exist for a specified application and reference code.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

Reference\$ The reference code for this specific application parameters

FUNCTION ExistsVariable(Application\$, Reference\$, VariableName\$, DataType\$)

Check to determine if a specific parameter exists for a specified application and reference code.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

Reference\$ The reference code for this specific application parameters

VariableName\$ The name of the parameter to locate.

DataType\$ The type of variable to be located.

FUNCTION AddVariable(Application\$, Reference\$, VariableName\$, DataType\$, MinimumValue\$, MaximumValue\$, Prompt\$, DefaultValue\$)

Add a new parameter value for a specified application and reference code. Only adds to the default entry if it does not already exist.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

Reference\$ The reference code for this specific application parameters

VariableName\$ The name of the parameter to locate.

DataType\$ The type of variable to be located.

MinimumValue\$ The minimum value to be accepted for the parameter

MaximumValue\$ The maximum value to be accepted for the parameter

Prompt\$ Description for the parameter

DefaultValue\$ The default value to be assigned to this parameter

FUNCTION RemoveVariable(Application\$, Reference\$, VariableName\$, DataType\$)

Delete a specific parameter from the file for a specified application and reference code.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

Reference\$ The reference code for this specific application parameters

VariableName\$ The name of the parameter to locate.

DataType\$ The type of variable to be located.

FUNCTION RemoveReference(Application\$, Reference\$)

Delete the parameters associated with a specific application and reference code. The will also delete all user/machine specific values as well.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

Reference\$ The reference code for this specific application parameters

FUNCTION RemoveApplication(Application\$)

Delete the parameters associated with a specific application. The will also delete parameters for every reference code and all user/machine specific values as well.

Parameter	Description
-----------	-------------

Application\$ The code used to identify the application

FUNCTION CloneReference(Application\$, SourceReference\$, TargetReference\$)

Duplicate all the parameters associated with a specific application and reference code to a second reference code. The will also duplicate all user/machine specific values as well.

Parameter	Description
Application\$	The code used to identify the application
SourceReference\$	The source reference code for the parameters
TargetReference\$	The target reference code where the parameters will be copied

FUNCTION CloneApplication(SourceApplication\$, TargetApplication\$)

Duplicate all the parameters associated with a specific application to a second application. The will also duplicate every reference code and all user/machine specific values as well.

Parameter	Description
SourceApplication\$	The source application code for the parameters
TargetApplication\$	The target application code where the parameters will be copied

Class *viewer/preview

This object contains the logic to create and control the GUI Print Previewer

Properties

PROPERTY Version\$

PROPERTY Language\$

Language to be used for message libraries

PROPERTY Report

A reference to the *viewer/report object that contains the content of the generated document.

Type [\[ProvideX Class: *viewer/report\]](#)

PROPERTY Software\$

Description of this application from the message library (default: Report Viewer)

PROPERTY SuppressSplashScreen

Boolean (0=False, 1=True)

PROPERTY SuppressStatusBar

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY SuppressToolBar

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY SuppressToolTips

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY SuppressWaterMarks

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY SuppressBannerInfo

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY Caption\$

Window caption of viewer

PROPERTY Icon\$

Icon for upper left corner of viewer

PROPERTY Left

Upper Left Location of primary Window X-Axis in pixels

PROPERTY Right

Width of primary Window in pixels

PROPERTY Top

Upper Left Location of primary Window Y-Axis in pixels

PROPERTY Bottom

Height of primary Window in pixels

PROPERTY Column

Location of base window on current monitor

PROPERTY Row

Location of base window on current monitor

PROPERTY Columns

Width of base window on current monitor

PROPERTY Rows

Height of base window on current monitor

PROPERTY SetWindowStateOnCreate

Boolean (0=False, 1=True) - set to True to enable

PROPERTY SetWindowStateOnSpoolStart

Boolean (0=False, 1=True) - set to True to enable

PROPERTY SetWindowStateOnFirstPage

Boolean (0=False, 1=True) - set to True to enable

PROPERTY SetWindowStateOnSpoolComplete

Boolean (0=False, 1=True) - set to True to enable

PROPERTY LookAndFeel

LookAndFeel and ThemeName must be set prior to doing an 'Init()' or changing it will have no effect

Value Description

0|1|2 Auto select

3 3D

4 4D

PROPERTY ThemeName\$

LookAndFeel and ThemeName must be set prior to doing an 'Init()' or changing it will have no effect

PROPERTY Visible

Set primary window to be visible (1=true, 0=false)

PROPERTY Page

The current page

PROPERTY Pages

Total number of pages in the document

PROPERTY View

Set the type of view to be used

Value	Description
0	1 page at a time, no page scrolling (Default)
1	2 pages at a time, no page scrolling (odd on left, even on Right)
2	2 pages at a time, odd on right, Even on Left (Book Style)
3	4 pages at a time

PROPERTY Zoom

Set the zoom per centage

PROPERTY FileName\$

PROPERTY DPIScreen

PROPERTY Printer\$

The last printer that was used

PROPERTY PrinterDescription\$

The description for the last printer used

PROPERTY DisableSaveAs

Boolean (0=False, 1=True) - set to True to disable

PROPERTY DisableDelete

Boolean (0=False, 1=True) - set to True to disable

PROPERTY DisablePrint

Boolean (0=False, 1=True) - set to True to disable

PROPERTY DisableSearch

Boolean (0=False, 1=True) - set to True to disable

PROPERTY DisableWatermarks

Boolean (0=False, 1=True) - set to True to disable

PROPERTY DisableBanner

Boolean (0=False, 1=True) - set to True to disable

PROPERTY Collate

Set to ZERO to disable collation (default = 1 - Collate)

PROPERTY Colour

Set to ZERO for monochrome output (default = 1 - colour)

PROPERTY Color

PROPERTY Range\$

Set range of pages in document to be viewed

Value	Description
ALL	Entire document (default)
StartPageNum:EndPageNum	For a range of pages
Page,Page,FromPage:ToPage,Page	For selected pages or page ranges

PROPERTY Copies

Number of copies to be printed (default = 1)

PROPERTY DPIPrinter

Printer resolution (0 = printer default)

PROPERTY DefaultOpenPath\$

PROPERTY DefaultSaveAsPath\$

PROPERTY AlwaysUseDefaultOpenPath

PROPERTY DefaultPaperSize

Set the default paper size (default of 0 = letter)

PROPERTY DefaultOrientation

Set the default orientation (default of 0 = portrait)

PROPERTY DefaultOrientation\$

PROPERTY DefaultFixedFont\$

Set the name for the default font (default = Courier New)

PROPERTY DefaultFixedFontSize

Set the default fixed font size (0 - calculate based on the report columns and lines)

PROPERTY MinimumColumns

Set the minimum number of columns for report ie: 132 columns

PROPERTY MinimumRows

Set the minimum number of lines for report ie: 66 lines (0=Auto calculate based on number of lines in first non-blank page)

PROPERTY PaginationAt

Set method used for pagination (0=Contains FormFeeds, >0 Line number to paginate at)

PROPERTY ScaleToFit

Set to 1 to force report to fit the page

PROPERTY WaterMark

This is a reference to a [\[ProvideX Class:watermark\]](#) object for the report

PROPERTY WaterMarkPages

Control where watermark is printed

Value	Description
--------------	--------------------

0	None
1	First Page
2	All Pages

PROPERTY OutputEvenNumberOfPages

Set to 1 to force all reports to produce an even number of Pages in total

PROPERTY Banner

Print a banner page

Value	Description
--------------	--------------------

0	No banner page
1	Single banner page before the job
2	A banner page before each copy

PROPERTY SuppressFirstBlankPage

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY SuppressAllBlankPages

Boolean (0=False, 1=True) - set to True to suppress

PROPERTY PageGutterVertical

Set page vertical gutter size

PROPERTY PageGutterHorizontal

Set page horizontal gutter size

PROPERTY HelpFile\$

Filename of Helpfile

PROPERTY HelpIndex\$

Reference for Helpfile Index page

PROPERTY HelpContents\$

Reference for the Helpfile Contents page

PROPERTY UsePrinter\$

Name of default printer for output

PROPERTY SuppressPrinterSelection

Set to ONE (1) for suppress selection of the printer

Methods

FUNCTION Print()

print whole report with user printer selection dialogue

FUNCTION Print(Printer\$)

print whole report to printer specified

FUNCTION Open()

Open a dialogue to select the document file to view

FUNCTION Open(FileName\$)

Open the specified document file

Parameter	Description
------------------	--------------------

FileName\$	The name of the document file to open
------------	---------------------------------------

FUNCTION Open(ObjectHandle)

Open the specified report in the viewer

Parameter	Description
------------------	--------------------

ObjectHandle	A reference to a [ProvideX Class: *viewer/report] that holds the report content
--------------	---

FUNCTION Close()

Close the document loaded in the viewer

FUNCTION SaveAs()

Show a SaveAs Dialogue

FUNCTION SaveAs(FileName\$)

force a save as to a specific name

FUNCTION SaveAsPDF()

Show a SaveAs Dialogue

FUNCTION SaveAsPDF(FileName\$)

force a save as to a specific name

FUNCTION Find()

show dialog to prompt for search text

FUNCTION Find(String\$)

Search for specified text in the document

FUNCTION Find(String\$, CaseSensitive)

Search for specified text in the document and also indicate if the search should be case sensitive

FUNCTION FindNext()

Find next occurrence of current search string

FUNCTION FindPrevious()

Find previous occurrence of current search string

FUNCTION Delete()

Close the current document and delete the file

FUNCTION Init()

Create the viewer window

FUNCTION Quit()

Close and exit the viewer

FUNCTION GotoPage()

Open a dialogue to prompt for the page to be displayed in the current document

FUNCTION GotoPage(Page%)

Display the specified page of the current document

Parameter	Description
Page%	The page number to display

FUNCTION FirstPage()

Display the first page of the current document

FUNCTION LastPage()

Display the last page of the current document

FUNCTION NextPage()

Display the next page of the current document

FUNCTION PreviousPage()

Display the previous page of the current document

FUNCTION ScrollUp()

Scroll up a single page

FUNCTION ScrollUp(Amount)

Scroll up a specified number of pages

Parameter	Description
Amount	The number of pages to scroll

FUNCTION ScrollDown()

Scroll down a single page

FUNCTION ScrollDown(Amount)

Scroll down a specified number of pages

Parameter	Description
Amount	The number of pages to scroll

FUNCTION ScrollLeft()

Scroll the current page to the left

FUNCTION ScrollLeft(Amount)

Scroll the current page to the left the specified number of characters

Value	Description
Amount	The number of characters to scroll

FUNCTION ScrollRight()

Scroll the current page to the right

FUNCTION ScrollRight(Amount)

Scroll the current page to the right the specified number of characters

Value	Description
Amount	The number of characters to scroll

FUNCTION Zoom()

Open a dialogue to allow the user to select the zoom level

FUNCTION Zoom(ZoomLevel%)

Set the zoom level to the specified value

Value	Description
ZoomLevel%	The new zoom level

FUNCTION Zoom(ZoomLevel%, ColumnPercent, RowPercent)

Set the zoom level to the specified value and center the page at specified point

Value	Description
ZoomLevel%	The new zoom level
ColumnPercent	Column for center of page
RowPercent	Row for center of page

FUNCTION ZoomIn()

Zoom in one notch

FUNCTION ZoomIn(ColumnPercent, RowPercent)

FUNCTION ZoomOut()

Zoom out one notch

FUNCTION ZoomOut(ColumnPercent, RowPercent)

FUNCTION Event(Text\$, ControlVal, EOMS, Mouse\$, UseTimer%)

FUNCTION Process()

FUNCTION SetCallBack(EventName\$, Routine\$)

Set the printer call back program to the specified routine

Parameter	Description
------------------	--------------------

EventName\$	"PRINTER"
-------------	-----------

Routine\$	The name of the program routine to be called
-----------	--

The developer specified call back routine for printer selection will be accessed using:

```
call Routine$, PrinterChan, ThePrinter$, Desc$, Options$,  
IsGUI, Title$
```

- PrinterChan - Channel for printer (0 if open failed)
- ThePrinter\$ - The name of the selected printer
- Desc\$ - Description of printer that is returned
- Options\$ - Options that are returned
- IsGUI - Boolean to indicate that the application is graphical
- Title\$ - Report title or application name if there is not title

FUNCTION View(ViewNumber)

FUNCTION RePaint()

Class *viewer/report

Properties

PROPERTY Version\$

PROPERTY DefaultPage

PROPERTY WaterMark

PROPERTY Page

PROPERTY Pages

PROPERTY Spooled

PROPERTY Spool

PROPERTY Title\$

PROPERTY FileName\$

PROPERTY FileName

PROPERTY SuppressFirstBlankPage

PROPERTY SuppressAllBlankPages

PROPERTY SuppressWatermarks

PROPERTY DefaultFixedFont\$

PROPERTY DefaultFixedFontSize

PROPERTY MinimumColumns

PROPERTY MinimumRows

PROPERTY PaginationAt

PROPERTY ScaleToFit

PROPERTY ReportType

PROPERTY Printer\$

PROPERTY DPI

PROPERTY ScreenDPI

PROPERTY Range\$

PROPERTY Collate

PROPERTY Copies

PROPERTY Colour

PROPERTY Color

PROPERTY Converter

PROPERTY Search

Methods

FUNCTION Search(Text\$, CaseSensitive, Direction, Page, XLocation, YLocation)

FUNCTION SearchReset()

FUNCTION GetInfo()

FUNCTION GetInfo(Item%, Name\$, Value\$)

FUNCTION Load()

FUNCTION Page()

FUNCTION Page(PageNum%)

FUNCTION RenderPage(Channel%)

FUNCTION RenderBlankPage(Channel%)

Class *viewer/splashscreen

Properties

PROPERTY Version\$

PROPERTY Visible

Set visibility of the splash screen

Value	Description
-------	-------------

0	Not visible (destroy if currently visible)
---	--

1	Visible
---	---------

PROPERTY Image\$

Path and Filename of image to use as splash

If running locally this is the name of the local image. If running in WindX, then this is the name of the image from WindX's point of view. If running in JavX, then this is the local (server side) image name, to get sizing.

PROPERTY JavXImage\$

Path and Filename of image for JavX to retrieve

PROPERTY Status

PROPERTY Status\$

PROPERTY StatusFont\$

PROPERTY StatusColour\$

Methods

No Methods

Class *viewer/watermark

This object contains the logic to create and control watermarks

Constructor

Properties

PROPERTY Version\$

PROPERTY Text\$

Text to be used for the watermark

PROPERTY TextLocation

Location of the watermark text on the page; allowed values are:

Value	Description
-------	-------------

0	Centered
1	Tiled
2	TopLeft
3	TopRight
4	BottomLeft
5	BottomRight

PROPERTY Font\$

Font to be used for the watermark text (IE: "Times Roman,-168,B")

PROPERTY Color\$

Colour to use for the watermark text

PROPERTY Colour\$

Colour to use for the watermark text

PROPERTY Rotation

Angle of rotation for the watermark text

PROPERTY PageNumbers

Override watermark and use the page number instead of text or an image.

Value	Description
-------	-------------

0	Use text and image as the watermark
1	Use the page number as the watermark; ignore the text and image

PROPERTY Image\$

Filename of an image to use as a watermark

PROPERTY ImageLocation

Location of the watermark image on the page; allowed values are:

Value	Description
0	Centered
1	Tiled
2	TopLeft
3	TopRight
4	BottomLeft
5	BottomRight

Methods

FUNCTION RenderWaterMark(ChannelNum%, PageNum%, PageWidth, PageHeight, DPI, MarginLeft, MarginTop, MarginRight, MarginBottom)

Create the watermark for the output document.

Parameter	Description
ChannelNum%	Channel number for the output device
PageNum%	Page number where the watermark is to be printed
PageWidth	Width of page (including left and right margins)
PageHeight	Height of page (including top and bottom margins)
DPI	Printing resolution for document
MarginLeft	Size of left margin
MarginTop	Size of top margin
MarginRight	Size of right margin
MarginBottom	Size of bottom margin