

## See Also

[GraffitiWebPDFTable](#)

[GraffitiWebPDFCellStyle](#)

## Enumerations

This class exposes no enumerations.

## Contants

Name	Value
FormatA3 as String	"a3"
FormatA4 as String	"a4"
FormatA5 as String	"a5"
FormatLegal as String	"legal"
FormatLetter as String	"letter"
LineStyleBevel as Number	0
LineStyleRound as Number	1
LineStyleSquare as Number	2
OrientationLandscape as String	"landscape"
OrientationPortrait as String	"portrait"
UnitCentimeters as String	"cm"
UnitInches as String	"in"
UnitMillimeters as String	"mm"
UnitPoints as String	"pt"

## Events

Definition	Description
PDFOutput( theContent as String )	Fired when the PDF content has been generated. theContent is a BASE64 compatible URI.
TableUpdated( theTable as GraffitiWebPDFTable)	Raised when a table has been added to the document and its StartY and EndY properties are populated.

## Methods

Definition	Description
AddBezier( StartX as Double, StartY as Double, _ X1 as Double, Y1 as Double, _ X2 as Double, Y2 as Double, _ X3 as Double, Y3 as Double )	Add a bezier curve to the PDF.
AddCircle( X as Double, Y as Double, _ Radius as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Add a circle to the PDF.

Definition	Description
AddEllipse( X as Double, Y as Double, _ RadiusX as Double, RadiusY as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Add an ellipse to the PDF.
AddHTML( HTMLContent as String, W as Integer = 1200 )	Add HTML to the PDF. W is the width of the area to render on.
AddImage( PictureData as String, Type as String, _ X as Double, Y as Double, _ W as Double = - 1, H as Double = - 1 )	Add an image to the PDF. PictureData should be a BASE64 compatible URI string.
AddLine( X1 as Double, Y1 as Double, _ X2 as Double, Y2 as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Draws a line on the PDF.
AddPage( W as Double = 210, H as Double = 297 )	Adds a page to the PDF. W and H are dimensions.
AddRect( X as Double, Y as Double, _ W as Double, H as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Draws a rectangle on the PDF.
AddRoundedRect( X as Double, Y as Double, _ W as Double, H as Double, _ RadiusX as Double, RadiusY as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Draws a rounded rectangle on the PDF.
AddTable( Data as GraffitiWebPDFTable )	Draws a table on the PDF.
AddText( theContent as String, X as Integer, Y as Integer, BlockWidth as Double = -1 )	Draws text on the PDF.
AddTriangle( X1 as Double, Y1 as Double, _ X2 as Double, Y2 as Double, _ X3 as Double, Y3 as Double, _ hasFill as Boolean = False, hasStroke as Boolean = False )	Draws a triangle on the PDF.
Download( Filename as String )	Constructs the PDF and sends it to the browser for download.
FromHTML( theHTML as String, DocumentOrientation as String = "portrait", _ DocumentUnits as String = "pt", DocumentFormat as String = "letter", _ Margin as Double = 0 )	Creates a PDF from an HTML string.
GetTable( tableID as String ) As GraffitiWebPDFTable	Gets an added table object.
LoadInViewer( theViewer as WebHTMLViewer, theContent as String )	Will load the output of the PDFOutput event in to a WebHTMLViewer.
NewDocument( DocumentOrientation as String = "portrait", _ DocumentUnits as String = "mm", _ DocumentFormat as String = "A4" )	Start a new PDF document.
Output()	Tell GraffitiWebPDF to compile and output the data.
Save() Deprecated	Tells GraffitiWebPDF to compile the PDF data and prompt the user for download. This method has been deprecated as it may not behave properly across all browsers.

Definition	Description
SetFillColor( theColor as Color )	Changes the document's Fill Color for items added after calling this method.
SetFillColor( Gray as Integer )	
SetFillColor( Red as Integer, Green as Integer, Blue as Integer )	
SetFillColor( Cyan as Integer, Magenta as Integer, Yellow as Integer, Key as Integer )	
SetFont( Face as String, Style as String )	Changes the document's font face and style for items added after calling this method.
SetFontSize( newSize as Double )	Changes the document's font size for items added after calling this method.
SetFontStyle( Style as String )	Changes the document's font style for items added after calling this method.
SetLineCap( Style as Integer )	Changes the style of line end caps.
SetLineJoin( Style as Integer )	Changes the style of line joints.
SetLineWidth( W as Double )	Sets the thickness of lines drawn after calling this method.
SetPage( PageNumber as Integer )	Changes the currently selected page within the document.
SetProperties( Title as String, Subject as String = "", _ Author as String = "", Keywords as String = "", _ Creator as String = "GraffitiWebPDF" )	Sets the properties of the current document.
SetStrokeColor( theColor as Color )	Changes the stroke color for items added to the document after this method is called.
SetStrokeColor( Gray as Integer )	
SetStrokeColor( Red as Integer, Green as Integer, Blue as Integer )	
SetStrokeColor( Cyan as Integer, Magenta as Integer, Yellow as Integer, Key as Integer )	
SetTextColor( theColor as Color )	Changes the text color of items added after this method is called.
SetTextColor( Gray as Integer )	
SetTextColor( Red as Integer, Green as Integer, Blue as Integer )	
SetTextColor( Cyan as Integer, Magenta as Integer, Yellow as Integer, Key as Integer )	

## Properties

Name	Type	Description
AvailableFonts	JSONItem	The fonts on the user's machine which are available for use in PDFs.

## Examples

### Saving PDFs on the Server

This is a simple example of saving PDF data to a file on the server. Note that this saves to the same place each time, and could potentially fail based on target and permissions. You should modify this to save each file separately in a safe place.

This code belongs in the PDFOutput Event Handler on your WebPage or WebContainer.

```
dim b64Content as String = theContent.NthField( ",", 2 )
dim f as FolderItem = GetFolderItem( "newPDF.pdf" )
if not IsNull( f ) then
    dim b as BinaryStream = BinaryStream.Create( f, True )
    b.Write( DecodeBase64( b64Content ) )
    b.Close
end if
```

## Notes

### HTML Methods

The AddHTML and FromHTML methods are currently experimental per the developers of the underlying library. Issues may arise depending on your usage of these methods.

In FromHTML, you can explicitly add your own page breaks by inserting `<div class='html2pdf__page-break'></div>` in your HTML.

Due to a BASE64 conversion and display issue in some browsers, PDF files created with FromHTML may not display in an HTMLViewer. These are, however, proper PDFs when downloaded.

From:

<https://graffitisuite.com/wiki/> - **GraffitiSuite Documentation**

Permanent link:

<https://graffitisuite.com/wiki/doku.php?id=web:pdf>

Last update: **2020/02/22 08:06**

