

23.13 DocumentView (gb.form)

Dieses Steuerelement implementiert einen Dokumentenbetrachter.

Die Klasse ist erzeugbar. So erzeugen Sie ein neues DocumentView-Objekt:

```
Dim hDocumentView As DocumentView
hDocumentView = New DocumentView ( Parent As Container ) [ As "EventName" ]
```

23.13.1 Eigenschaften

Die Klasse DocumentView verfügt über diese Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Arrangement	Integer	Gibt zurück oder legt fest, wie die verschiedenen Seiten innerhalb der Ansicht angeordnet sind. Diese Eigenschaft kann einen der folgenden Werte annehmen: Arrange.Vertical, Arrange.Horizontal, Arrange.Row, Arrange.Column oder Arrange.Fill.
AutoCenter	Boolean	Gibt zurück oder legt fest, ob die Seite zentriert ist, wenn nur <u>eine</u> Seite in der Breite angezeigt wird. Dies ist nur bei einigen Anordnungsmodi verfügbar: Arrange.Vertical, Arrange.Horizontal oder Arrange.Fill.
Center	Boolean	Rückgabe oder Festlegung, ob die Seite in der Ansicht zentriert wird oder nicht.
Border	Boolean	Rückgabe oder Festlegung, ob die Dokumentenansicht einen Rahmen hat.
Column	Integer	Damit können Sie die Anzahl der angezeigten Spalten, abhängig vom gewählten Anordnungsmodus, festlegen.
Columns	Integer	Rückgabe oder Festlegung der Anzahl der Spalten, die für das Layout der Seiten verwendet werden. Ist dieser Wert auf Null gesetzt (Standard), so werden die Seiten in der Vorschau nur einzeln angezeigt. Andernfalls werden alle Seiten von oben nach unten angeordnet.
Count	Integer	Rückgabe oder Festlegung der Anzahl der Seiten des Dokuments.
Desaturate	Boolean	Rückgabe oder Festlegung, ob die Vorschau automatisch entsättigt wird.
Direction	Integer	Rückgabe oder Festlegung der Richtung des Textes. Sie kann einen der folgenden Werte annehmen: Direction.Default, Direction.RightToLeft, Direction.LeftToRight.
DocHeight	Integer	Gibt die Standardhöhe einer Seite in Millimetern zurück oder legt sie fest. Die Standard-Höhe der Seite ist 297-mm (A4-Format).
DocWidth	Integer	Gibt die Standardbreite einer Seite in Millimetern zurück oder legt sie fest. Die Standard-Breite der Seite ist 210-mm (A4-Format).
FirstVisibleDocument	Integer	Gibt die Nummer der ersten sichtbaren Seite im Viewer zurück.
LastVisibleDocument	Integer	Gibt die Nummer der letzten sichtbaren Seite im Viewer zurück.
FirstVisiblePage	Integer	Gibt den Index der ersten sichtbaren Seite im Viewer zurück.
LastVisiblePage	Integer	Gibt den Index der letzten sichtbaren Seite im Viewer zurück.
Layout	_DocumentLayout	Gibt ein virtuelles Objekt zurück, das für die Definition des Layouts einer bestimmten Seite verwendet wird. Siehe Ereignis 'Layout' für weitere Details. Die Klasse _DocumentLayout hat 2 Eigenschaften: (1) Height: Gibt die Höhe der Seite in Millimeter an oder legt sie fest und (2) Width: Gibt die Breite der Seite in Millimetern zurück oder legt sie fest.
Max	Integer	Gibt die Anzahl der Seiten des Dokuments - vermindert um 1 – zurück und entspricht so dem höchsten Seiten-Index.
Padding	Integer	Rückgabe oder Festlegung des (inneren) Randabstands innerhalb der Ansicht in Millimetern. Die Angabe gilt für <u>alle</u> 4 Richtungen!
PageHeight	Integer	Rückgabe oder Festlegung der Standardhöhe einer Seite in Millimetern. Die Standardgröße der Seite ist 210-mm x 297-mm (A4-Format).

Eigenschaft	Datentyp	Beschreibung
PageWidth	Integer	Rückgabe oder Festlegung der Standardbreite einer Seite in Millimetern. Die Standardgröße der Seite ist 210-mm x 297-mm (A4-Format).
ShowPage	Boolean	Rückgabe oder Festlegung, ob die Seitenzahl innerhalb der Ansicht angezeigt werden soll.
ShowShadow	Boolean	Rückgabe oder Festlegung, ob die Seiten mit einem kleinen Schatten gezeichnet werden sollen.
Zoom	Float	Gibt den Zoom-Faktor der Ansicht zurück oder legt ihn fest. 'Zoom' ist ein Synonym für die Eigenschaft 'Scale'.

Tabelle 23.13.1.1 : Eigenschaften der Klasse DocumentView

23.13.2 Methoden

Die Klasse *DocumentView* verfügt über diese wesentlichen Methoden:

Methode	Beschreibung
Find (X As Integer, Y As Integer)	Gibt die Seitenzahl unter einer bestimmten Mauszeigerposition zurück. X und Y sind die Positionen des Mauszeigers relativ zur Dokumentansicht.
Goto (Page As Integer)	Verschiebt die Ansicht auf eine bestimmte Seite. 'Page' ist die Seitennummer, beginnend bei Null.
Refresh ()	Aktualisiert den Inhalt der Ansicht im Viewer.
Reset ()	Setzt die Ansicht zurück.
Show ()	Zeigt das Steuerelement an.

Tabelle 23.13.2.1 : Methoden der Klasse DocumentView

23.13.3 Ereignisse

Die Klasse *DocumentView* verfügt über diese Ereignisse:

Ereignis	Beschreibung
Draw (Page As Integer, Width As Integer, Height As Integer)	Dieses Ereignis wird ausgelöst, wenn eine bestimmte Seitenvorschau gezeichnet werden muss: 'Page' ist der Index der zu zeichnenden Seite. 'Width' und 'Height' sind die Größe des Bildpuffers, in dem die Seite gezeichnet werden soll.
Finished ()	Dieses Ereignis wird ausgelöst, wenn alle Seitenvorschauen gezeichnet worden sind.
Layout (Page As Integer)	Dieses Ereignis wird ausgelöst, wenn sich der Parameter Page ändert und damit jedes Seitenlayout unabhängig voneinander definiert werden kann. 'Page' ist der Index der zu gestaltenden Seite. Verwenden Sie die Eigenschaft Layout innerhalb des Handlers dieses Ereignisses, um das Layout der *angegebenen* Seite zu definieren.
Zoom ()	Dieses Ereignis wird ausgelöst, wenn sich die Zoom-Stufe geändert hat.

Tabelle 23.13.3.1 : Ereignisse der Klasse DocumentView

23.13.3.1 Anwendungsmöglichkeiten einer DocumentView

Der Entwickler des Steuerelementes Fabien Bodard beschrieb die Anwendungsmöglichkeiten einer DocumentView so: Sie kann nicht nur PDF-Dokumente anzeigen, sondern eigentlich alles, was Sie möchten, solange es ein Dienstprogramm gibt, mit dem Sie den Inhalt rendern können. DocumentView liefert Ihnen 'Leinwände', auf denen Sie das zeichnen können, was Sie wollen. Es kann also alles Mögliche sein, solange Sie es mit der Funktion Paint.DrawImage() zeichnen können.

23.13.3.2 Beispiel 1

Eine typische Anwendung für das Steuerelement DocumentView ist die Anzeige von PDF-Dateien:



Abbildung 23.13.3.2.1: Anzeige eines PDF-Dokumentes

Quelltext:

```
' Gambas class file

Private $PDF_Document As New PdfDocument

Public Sub Form_Open()
    btnOpen.SetFocus()
End

Public Sub btnOpen_Click()
    Dialog.Filter = [ "*.pdf", "PDF-Files" ]
    If Not Dialog.OpenFile() Then
        $PDF_Document.Open(Dialog.Path)
        '-- Setup the number of pages shown in the viewer. It enforce the layout event call
        DocumentView1.Count = $PDF_Document.Count
        DocumentView1.Refresh()
        Wait
    Else
        Return
    Endif
End

Public Sub DocumentView1_Layout(Page As Integer)
    DocumentView1.Layout.Width = $PDF_Document[Page + 1].Width
    DocumentView1.Layout.Height = $PDF_Document[Page + 1].Height
End

Public Sub DocumentView1_Draw(Page As Integer, Width As Integer, Height As Integer)
    '-- Draw the pdf image
    Paint.DrawImage($PDF_Document[Page + 1].Image, 0, 0, Paint.Width, Paint.Height)
End

Public Sub DocumentView1_Zoom()
    $PDF_Document.Zoom = Last.Zoom
End

Public Sub Form_Resize()
    DocumentView1.Column = 1
    DocumentView1.Refresh()
    Wait
End

Public Sub btnClose_Click()
    FMain.Close()
End
```

23.13.3.3 Beispiel 2

Im zweiten Projekt wird jeder Bild-Inhalt auf ein separates Image – als Leinwand – gezeichnet und in der Größe angezeigt, die durch den Schieberegler im Bereich von 0% bis 200% für jede einzelne Seite vorgegeben wird:

```
Public Sub DocumentView1_Draw(Page As Integer, Width As Integer, Height As Integer)
    Paint.DrawImage($aImages[Page], 0, 0, Width, Height)
End
```

Dabei steht nicht die detaillierte Vorschau einzelner Bilder im Vordergrund, sondern die Übersicht zu allen Bildern im ausgewählten Verzeichnis:



Abbildung 23.13.3.3.1: Verkleinerte Anzeige von Bildern in einer DocumentView

Quelltext:

```
' Gambas class file

Private $sPath As String
Private $aImages As New Image[]

Public Sub Form_Open()

    lblMin.Text = sliderZoom.MinValue
    lblMax.Text = Str(sliderZoom.MaxValue) & "%"

    DocumentView1.Arrangement = Arrange.None
    DocumentView1.Columns = 0
    DocumentView1.Center = True
    DocumentView1.Spacing = 20

    sliderZoom.Value = 30
    DirBox1.Tooltip = "Select an image directory!"
    ChangePath()

End

Public Sub DirBox1_Change()
    $sPath = Last.Path
    ChangePath()
    sliderZoom.Value = 32
End

Public Sub ChangePath()

    Dim sFile As String
    Dim hImage As Image

    $aImages.Clear()
    For Each sFile In Dir($sPath, Null, gb.File)
        Select Case File.Ext(sFile)
            Case "png", "jpg"
                hImage = Image.Load($sPath & "/" & sFile)
                $aImages.Add(hImage)
        End Select
    Next
    DocumentView1.Count = $aImages.Count
End
```

```

Public Sub DocumentView1_Draw(Page As Integer, Width As Integer, Height As Integer)
    Paint.DrawImage($aImages[Page], 0, 0, Width, Height)
End

Public Sub DocumentView1_Layout(Page As Integer)
    '-- Allow to define specific layout for items
    '-- It is called before the drawing... so images need to be seen twice in this case
    DocumentView1.Layout.Width = $aImages[Page].W
    DocumentView1.Layout.Height = $aImages[Page].H
End

Public Sub DocumentView1_Zoom()
    DocumentView1.Refresh()
End

Public Sub sliderZoom_Change()
    DocumentView1.Zoom = sliderZoom.Value / 100
    lblMax.Text = Str(sliderZoom.Value) & "%"
End

```

Sie finden die Projekt-Archive für die beiden vorgestellten Projekte im Download-Bereich.