

## 11.8 Programm-Hilfen

Wenn Sie ein umfangreiches Programm entwickelt und getestet haben, dann kennen Sie die Programm-Funktionen oder die Konfigurationseinstellungen am besten. Spendieren Sie deshalb dem Programm und damit allen zukünftigen Nutzern eine Programm-Hilfe. Die Hilfe sollten Sie so einrichten, dass diese über die Funktionstaste *F1* im Hauptprogramm aufgerufen werden kann und mit der Escape-Taste (ESC) wieder beendet wird. Ob Sie sich für eine einfache Textdatei oder eine umfangreiche Hilfe im HTML-Format oder im PDF-Format entscheiden, wird wesentlich durch die Art und die Komplexität des Programms mit bestimmt.

In den Beispielen wird davon ausgegangen, dass die *Hilfedatei* in unterschiedlichen Formaten im Projektordner liegt. Damit wird diese Datei mit in das Programm compiliert.

### 11.8.1 Aufruf der Hilfe im Hauptprogramm

Die Hilfe im Hauptprogramm wird aufgerufen, indem nach einem Druck auf die Taste *F1* das Ereignis *Form\_KeyPress()* im *Hauptprogramm* ausgewertet wird und als *Reaktion* ein weiteres Formular – hier *FHelp* – mit dem Hilfetext angezeigt wird:

```
PUBLIC SUB Form_KeyPress()
  IF Key.Code = Key.F1 THEN
  ' IF Key.Code = Key.F1 THEN AND FHelp.Closed = FALSE
    FHelp.Show
  ENDIF
  IF Key.Code = Key.ESC AND FHelp.Closed = FALSE THEN
    FHelp.Delete
  ENDIF
END ' Form_KeyPress()
```

Wenn das Hilfenfenster angezeigt wird und die Escape-Taste gedrückt wurde, dann wird das Hilfenfenster geschlossen.



Abbildung 11.8.1.1: Hauptprogramm

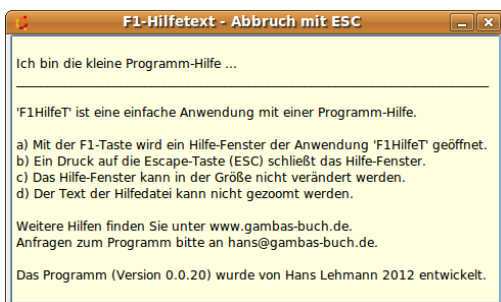


Abbildung 11.8.1.2: Hilfenfenster (Format Text/Plain)

### 11.8.2 Hilfe-Fenster – Textdatei

- Der Hilfetext im Format *Text/Plain* wird in eine *TextArea* als Anzeige-Objekt eingelesen. Stellen Sie sicher, dass Sie die Eigenschaft *TextArea1.ReadOnly* auf *TRUE* gesetzt haben.
- Sie haben keine Möglichkeit, den Hilfetext in besonderer Weise auszuzeichnen.

Im *Form\_KeyPress()-Ereignis* im Formular *FHelp* wird der *Key.Code* der Escape-Taste ausgewertet und mit dem Zerstören des Objekts 'Hilfenfenster' reagiert.

Da das Programm für das Hilfenfenster nur sehr kurz ist, wird der hier komplette Quelltext angegeben:

```
' Gambas class file

PUBLIC SUB Form_Open()
  FHelp.X = 200 ' u.U. anpassen...
  FHelp.Y = 200
  FHelp.Text = "F1-Hilfetext - Abbruch mit ESC"
  TextAreal.ReadOnly = TRUE
END ' Form_Open()

PUBLIC SUB Form_Show()
  FHelp.Arrangement = Arrange.Fill
  TextAreal.Background = &H00FFFFDF&
  TextAreal.Text = File.Load("help.txt")
END ' Form_Show()

PUBLIC SUB Form_KeyPress()
  IF Key.Code = Key.ESC AND FHelp.Closed = FALSE THEN
    FHelp.Delete
  ENDIF
END ' Form_KeyPress()
```

### 11.8.3 Hilfe-Fenster – HTML-Datei

- Liegt die Hilfedatei im HTML-Format vor, dann benötigen Sie als Anzeige-Objekt eine Browser-Komponente.

Quelltext:

```
' Gambas class file

PUBLIC SUB Form_Open()
  FHelp.W = 560
  FHelp.H = 238
  FHelp.X = 20 ' u.U. anpassen...
  FHelp.Y = 20
  FHelp.Text = "F1-Hilfetext - Abbruch mit ESC"
  WebBrowser1.X = 7
  WebBrowser1.Y = 7
END ' Form_Open()

PUBLIC SUB Form_Show()
  WebBrowser1.Path = Application.Path &/ "help.html"
END ' Form_Show()

PUBLIC SUB Form_KeyPress()
  IF Key.Code = Key.ESC AND FHelp.Closed = FALSE THEN FHelp.Delete
END ' Form_KeyPress()

PUBLIC SUB Form_Resize()
  WebBrowser1.W = FHelp.W - 11
  WebBrowser1.H = FHelp.H - 11
END ' Form_Resize()
```

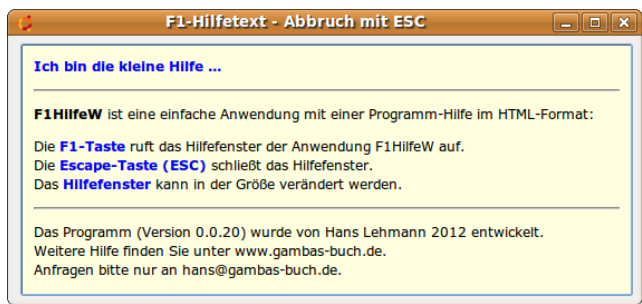


Abbildung 11.8.3.1: Hilfefenster (HTML-Format)

### 11.8.4 Hilfe-Fenster – PDF-Datei

- Wenn die Hilfedatei als PDF-Dokument vorliegt, ist der Programmabschnitt zur Anzeige der Hilfe umfangreicher. Der Aufwand lohnt vor allem für größere Projekte. Dann können Sie die Hilfe zum Beispiel komfortabel mit *LibreOffice Writer* schreiben, gestalten und abschließend das Dokument sofort in das PDF-Format konvertieren.

- Es wird der komplette Quelltext von *FHelp.class* angezeigt. Das vollständige Projekt finden Sie im Download-Bereich.
- Die Prozedur zum *Rendern einer PDF-Seite* ist einem Programm von *Daniel Campos Fernández* entnommen.
- Die Komponente *gb.pdf* müssen Sie über die Projekteigenschaften in das Projekt aufnehmen.

Quelltext:

```
' Gambas class file
Public pdfDokument As New PdfDocument
Public cIndex As Integer
Public CurrentPage As Integer
Public CurrentZoom As Float

Public Sub Form_Open()
    FHelp.H = 365
    FHelp.W = 650
    FHelp.X = 50 ' u.U. Koordinaten anpassen
    FHelp.Y = 50
    HBox1.H = 24
End ' Form_Open()

Public Sub Form_KeyPress()
    If Key.Code = Key.ESC And FHelp.Closed = False Then
        FHelp.Delete
    Endif ' KeyCode ?
End ' Form_KeyPress()

Public Sub Form_Show()
    pdfDokument.Close()
    Try pdfDokument.Open(Application.Path & "help.pdf")
    If Error Then
        Message.Error(Error.Text)
        Return
    Endif ' ERROR ?

    CurrentZoom = 1
    CurrentPage = 1

    RenderPage()

    btnPrev.Enabled = False
    If pdfDokument.Count > 1 Then
        btnNext.Enabled = True
    Else
        btnNext.Enabled = False
    Endif ' pdfDokument.Count > 1 ?
    btnZoomPlus.Enabled = True
    btnZoomMinus.Enabled = True
    btnRotate.Enabled = True
    picBox.Visible = True
End 'Form_Show()

Public Sub RenderPage()
    Dim hPic As Picture
    Dim iCount As Integer

    pdfDokument.Zoom = CurrentZoom
    ViewPort.Scroll(0, 0)
    hPic = pdfDokument[CurrentPage].Image.Picture

    Draw.Begin(hPic)
    For iCount = 0 To pdfDokument[CurrentPage].Result.Count - 1
        Draw.Rect(pdfDokument[CurrentPage].Result[iCount].Left,
            pdfDokument[CurrentPage].Result[iCount].Top,
            pdfDokument[CurrentPage].Result[iCount].Width,
            pdfDokument[CurrentPage].Result[iCount].Height)
    Next ' iCount
    Draw.End()

    picBox.Picture = hPic
    picBox.Resize(pdfDokument[CurrentPage].Width, pdfDokument[CurrentPage].Height)
    Form_Resize()
End ' RenderPage()

Public Sub Form_Resize()
    If CurrentPage = 0 Then Return
    picBox.Left = (ViewPort.Width - pdfDokument[CurrentPage].Width) / 2
End ' Form_Resize()
```

```
Public Sub btnNext_Click()
    Inc CurrentPage
    If CurrentPage = pdfDokument.Count Then btnNext.Enabled = False
    btnPrev.Enabled = True
    RenderPage()
End ' btnNext_Click()

Public Sub btnPrev_Click()
    Dec CurrentPage
    If CurrentPage = 1 Then btnPrev.Enabled = False
    btnNext.Enabled = True
    RenderPage()
End ' btnPrev_Click()

Public Sub btnZoomPlus_Click()
    If CurrentZoom < 3 Then CurrentZoom += 0.1
    If CurrentZoom = 3 Then btnZoomPlus.Enabled = False
    btnZoomMinus.Enabled = True
    RenderPage()
End ' btZoomIn_Click()

Public Sub btnZoomMinus_Click()
    If CurrentZoom > 0.5 Then CurrentZoom -= 0.1
    If CurrentZoom = 0.5 Then btnZoomMinus.Enabled = False
    btnZoomPlus.Enabled = True
    RenderPage()
End ' btZoomOut_Click()

Public Sub btnRotate_Click()
    Select Case pdfDokument.Orientation
        Case PdfDocument.Normal
            pdfDokument.Orientation = PdfDocument.Sideways
        Case PdfDocument.Sideways
            pdfDokument.Orientation = PdfDocument.Inverted
        Case PdfDocument.Inverted
            pdfDokument.Orientation = PdfDocument.SidewaysInverted
        Case PdfDocument.SidewaysInverted
            pdfDokument.Orientation = PdfDocument.Normal
    End Select ' pdfDokument.Orientation
    RenderPage()
End ' Rotation

Public Sub Form_Close()
    pdfDokument.Close()
End ' Form_Close()
```



Abbildung 11.8.4.1: Hilfenfenster (PDF-Format)

### 11.8.5 Hinweise

- Wo das Hilfenfenster angezeigt werden soll, wird durch die Größe und Lage des Hauptprogramm-Fensters bestimmt. Geben Sie keine Koordinaten vor, dann finden Sie das Hilfenfenster in der linken oberen Bildschirmcke mit den Standardwerten FHelp.X = 0 und FHelp.Y = 0.

Nimmt man als Kriterium die Art und Anzahl der Programm-Fenster, kann man bei einem Gambas-Projekt folgende Typen unterscheiden:

- Projekt mit genau einem Einzelfenster
- Projekt mit mehreren Einzelfenstern

- MDI-Anwendung als Projekt mit einem Haupt-Fenster und einzelnen Fenstern in der Container-Komponente *Workspace*
- Projekt mit genau einem Einzelfenster und mindestens einem eingebetteten, jedoch extern gestartetem Programm (*Embedder*).

In allen Fällen kann man die Programm-Hilfe nach dem o.a. Verfahren einfügen. Bei der MDI-Anwendung mit der Containerkomponente *Workspace* können Sie jedes Teil-Programm mit einer eigenen Hilfe versehen.