

17.13 DateChooser

Die Komponente DateChooser (gb.form) implementiert einen Kalender. Dieser Kalender ermöglicht es dem Benutzer, ein Datum in einem Kalender auszuwählen. Die Anzeige-Eigenschaften für ein ausgewähltes Datum im DateChooser können über die Eigenschaft *Data* und über das *Data-Ereignis* definiert werden. Die ideale Breite einer DateChooser-Komponente beträgt mindestens 296 Pixel, weil Sie nur dann Zugriff auf *alle* angebotenen Auswahl- und Navigationsmöglichkeiten haben.



Abbildung 17.13.1: Komponente DateChooser und DateBox (Anzeige)

Der Komponente *DateBox* wurde kein eigenes Kapitel gewidmet. Die Klasse *DateBox* enthält nur eine *ButtonBox* → Kapitel 16.8 *ButtonBox*, deren Maske auf Datums- und Zeitangaben gesetzt wurde und einen Dialog mit *DateChooser* öffnet. Mit dem Einsatz der Komponente *DateChooser* sind Sie auf der sicheren Seite, wenn es um die Verwendung *valider* Werte für ein Datum oder Zeitangaben (im 30-Minuten-Raster) in einem Gambas-Projekt geht. Wenn es klein und fein sein soll, dann kommen Sie um den Einsatz einer *DateBox* nicht herum.

17.13.1 Eigenschaften und Methoden DateChooser

Ausgewählte Eigenschaften werden in der folgenden Tabelle aufgeführt und beschrieben:

Eigenschaft	Typ	Default	Beschreibung
Value	Date	Now()	Gibt das selektierte Datum zurück
Day	Integer	-	Gibt den Tag des ausgewählten Datums zurück.
Month	Integer	-	Gibt den Monat des ausgewählten Datums zurück.
Year	Integer	-	Gibt das Jahr des ausgewählten Datums zurück.
Mode	Integer	0	Legt fest, ob im Kalender ein Datum, ein Datum und eine Zeit oder nur Zeiten angezeigt werden (Konstanten: <i>DateOnly</i> (0), <i>DateTime</i> (1) oder <i>TimeOnly</i> (2)).
Data	<i>_DateChooser_Data</i>	-	Diese virtuelle Klasse ermöglicht die Definition, wie man ein bestimmtes Datum formatiert. Die Eigenschaften müssen innerhalb des <i>Data-Event-Handlers</i> gesetzt werden.
Colors	<i>_DateChooser_Colors</i>	-	Diese virtuelle Klasse ermöglicht die Farb-Definition, wie man ein bestimmtes Datum farblich formatiert.

Tabelle 17.13.1.1: Ausgewählte Eigenschaften der Komponente DateChooser

Die virtuelle Klasse *_DateChooser_Data* definiert:

- Background: Der Standardwert ist *Color.Default*, das bedeutet, dass keine spezifische Hintergrundfarbe verwendet wird.
- Foreground : Der Standardwert ist *Color.Default*, das bedeutet, dass keine spezifische Vordergrundfarbe verwendet wird.
- Font: Der Standardwert (NULL) bedeutet, dass keine bestimmte Schriftart verwendet wird.
- Text: *Diese Eigenschaft wird gegenwärtig nicht benutzt.*

Die Klasse *DateChooser* besitzt mit *DateChooser.SetFocus()* nur *eine* ausgewiesene, spezielle Methode, die den Fokus auf die *DateChooser*-Komponente setzt.

17.13.2 Ereignisse DateChooser

Spezielle Ereignisse der Komponente *DateChooser* finden Sie hier:

Ereignis	Beschreibung
Activate	Wird ausgelöst, wenn der Benutzer doppelt auf ein Datum im DateChooser klickt.
Cancel	Wird ausgelöst, wenn der Benutzer die ESC-Taste gedrückt hat, um seine Auswahl aufzuheben.
Change	Wird ausgelöst, wenn das Datum im Kalender geändert wird.
Data(Date As Date)	Dieses Ereignis wird ausgelöst, um herauszufinden, wie ein bestimmtes Datum im Kalender angezeigt werden soll. Füllen Sie die Data-Eigenschaft mit den erforderlichen Informationen.

Tabelle 17.13.2.1: Übersicht zu den 4 speziellen Ereignissen der Komponente DateChooser

17.13.3 Projekt 1 DateChooser

Ausgewiesene Schwerpunkte im vorgestellten Projekt liegen einerseits darin, Ihnen die Verwendung der 2 virtuellen Klassen mit ihren Eigenschaften und Methoden im Zusammenhang mit dem Ereignis *DateChooser1_Data(Value As Date)* vorzustellen und andererseits zu zeigen, wie Sie ein ausgewähltes Datum im Kalender in einer Variable speichern oder es in geeigneten Komponenten anzeigen können.

Die Abbildung 17.13.1 zeigt die Programm-Oberfläche. Der Quelltext zum Projekt 1 wird vollständig angegeben und ausführlich kommentiert:

```
[1] ' Gambas class file
[2]
[3] Public dDatum As Date
[4]
[5] Public Sub Form_Open()
[6]     FMain.Center()
[7]     FMain.Resizable = False
[8] ' Mode-Konstanten stehen in der DateChooser-Klasse und nicht in der DateBox-Klasse!
[9]     DateBox1.Mode = DateChooser.DateOnly
[10]    DateChooser1.Mode = DateChooser.DateOnly
[11]    DateBox1.Value = Now()
[12]    GetDateValues()
[13]
[14]    DateChooser1.Colors["9/21/2013"] = Color.Orange ' Datum in englischer Notation!
[15]    DateChooser1.Colors["9/11/2013"] = Color.Yellow
[16]    DateChooser1.Colors["9/24/2013"] = Color.Red
[17]
[18]End ' Form_Open()
[19]
[20]Public Sub btnClearColors_Click()
[21]    DateChooser1.Colors.Clear()
[22]    DateChooser1.SetFocus()
[23]End ' btnClearColors_Click()
[24]
[25]Public Sub DateChooser1_Activate() ' Einfach-Klick
[26]    DateChooser1_Change()
[27]End ' DateChooser1_Activate()
[28]
[29]Public Sub DateChooser1_Change()
[30]    GetDateValues()
[31]    DateBox1.Value = DateChooser1.Value
[32]End ' DateChooser1_Change()
[33]
[34]Public Sub DateBox1_Activate() ' Doppel-Klick!
[35]    DateBox1_Change
[36]End ' DateBox1_Activate()
[37]
[38]Public Sub DateBox1_Change()
[39]    GetDateValues()
[40]    DateChooser1.Value = DateBox1.Value ' Synchronisation DateChooser -> DateBox
[41]End ' DateBox1_Change()
```

```

[42]
[43] Private Sub getDateValues()
[44]     Dim aWochentage As String[]
[45]
[46]     dDatum = DateChooser1.Value
[47]
[48]     txtDate.Text = Format(DateChooser1.Value, "dd.mm.yyyy")
[49]     txtDate2.Text = Format(DateChooser1.Value, "dd. mmm yyyy")
[50]     txtDate3.Text = Format(DateChooser1.Value, "d. mmmm yyyy")
[51]
[52]     txtDateDay.Text = DateChooser1.Day
[53]     txtDateMonth.Text = DateChooser1.Month
[54]     txtDateYear.Text = DateChooser1.Year
[55]
[56]     aWochentage=["Sonntag","Montag","Dienstag","Mittwoch","Donnerstag","Freitag","Samstag"]
[57]     txtDateDayOfWeek.Text = aWochentage[WeekDay(DateChooser1.Value)] ' Sonntag = 0
[58]     txtDateDayOfWeek.Text = Format(DateChooser1.Value, "dddd")
[59]     txtDateWeekNumber.Text = Week(DateChooser1.Value, gb.Monday, False)
[60]
[61] End ' getDateValues()
[62]
[63] Public Sub DateChooser1_Data(dValue As Date) ' dValue frei gewählter Variablenname
[64]
[65] ' If Month(dValue) = 12 And If Day(dValue) = 7 Then ' Genau 1 Wert
[66] ' If Day(dValue) = 12 Then ' Genau 12 Werte
[67] If WeekDay(dValue) = gb.Friday Then ' Mehr als 50 Werte ...
[68]     DateChooser1.Data.Background = &HC3DDFF
[69]     DateChooser1.Data.Foreground = Color.Red
[70]     DateChooser1.Data.Font = Font["Italic"]
[71] Endif ' WeekDay(dValue) = gb.Friday
[72]
[73] End ' DateChooser1_Data(dValue As Date)
[74]
[75] Public Sub btnClose_Click()
[76]     FMain.Close
[77] End ' btnClose_Click()

```

Kommentare:

- In der Zeile 3 wird eine projektweit geltende Variable *dDatum* vom Datentyp *Date* deklariert, der in der Zeile 46 das im Kalender ausgewählte Datum zugewiesen wird.
- Der Komponente *DateBox* muss ein Startwert (Zeile 11) mitgegeben werden, indes die Komponente *DateChooser* automatisch mit dem aktuellen System-Datum initialisiert wird.
- Die Methode *DateChooser.Colors.Clear* der virtuellen Klasse *_DateChooser_Colors* setzt den Wert für die Hintergrundfarbe (Zeile 21) auf den Standard-Wert der Komponente *ColorChooser* zurück, die zum Beispiel mit *DateChooser1.Colors["9/21/2013"] = Color.Orange* für das Datum 21.9.2013 gesetzt wurde → Zeile 14.
- Während der Variablen *dDatum* das Datum vom Typ *Date* zugewiesen wird, erfordern die meisten (Anzeige-)Komponenten – bis auf die *DateBox* – die Konvertierung vom Datentyp *Date* in den Datentyp *String*. In der selbst definierten Prozedur *getDateValues()* in den Zeilen 43 bis 61 werden sowohl das komplette Datum, der Tag, der Monat und das Jahr sowie der Wochentag und die Kalenderwoche in geeigneter Weise formatiert, wobei die *Format*-Funktion die größte Vielfalt bietet.
- In der Zeile 59 wird die Kalenderwoche aus dem selektierten Datum (1. Argument) berechnet und die folgenden Argumente sind optional. Durch die Angabe des 2. Argumentes wird der erste Tag der Woche definiert und mit dem 3. Argument die Zählweise für die Kalenderwochen. Dabei gilt: Wenn *FullWeek False* ist, dann wird die erste Teil-Woche des Jahres mitgezählt, wenn sie mindestens vier Tage im (neuen) Jahr hat.
- In den Zeilen 63 bis 73 wird im Event *DateChooser1_Data(Date As Date)* festgelegt, wie bestimmte Datumswerte formatiert werden sollen. Achtung: Sie müssen für den (symbolischen) Namen der Variablen vom Datentyp *Date* einen frei wählbaren Namen festlegen, der von *Date* verschieden ist, weil es sonst eine Fehlermeldung gibt! Mit dem Variablennamen *dValue* liegen Sie goldrichtig.
- In den Zeilen 68, 69 und 70 benutzen Sie 3 ausgewählte Eigenschaften der virtuellen Klasse *_DateChooser1_Data* zur Festlegung des Formats der Datumswerte, die auf die Auswahl – hier sind es *Freitage* – in der Zeile 67 passen → Abbildung 17.13.1.

17.13.4 Projekt 2 DateChooserT

Wenn Sie die Eigenschaft *DateChooser.Mode* auf *DateChooser.DateTime* setzen, ist die Komponente DateChooser die erste Wahl für die Festlegung von Terminen mit Datum und Uhrzeit:

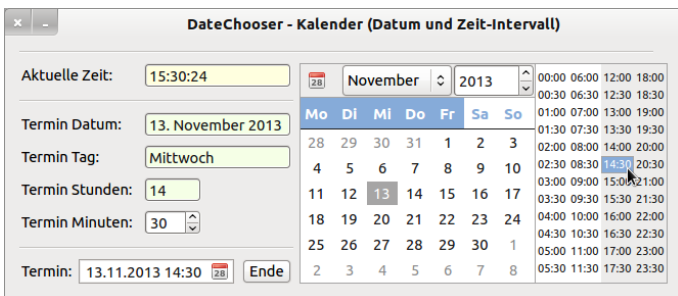


Abbildung 17.13.4.1: Komponente DateChooser – Termin 1

Das *Datum* wählen Sie mit einem (Einfach-)Klick und den Beginn für das *Zeitintervall* mit einem Doppelklick aus. Zusätzlich können Sie in diesem Projekt den Termin *minutengenau* festlegen, wenn das erforderlich ist. Der ausgewählte Termin (Datum, Beginn Zeitintervall, Minuten in der SpinBox (optional)) wird in einer DateBox angezeigt und in der Variablen *dEvent* vom Datentyp *Date* gespeichert.



Abbildung 17.13.4.1: Komponente DateChooser – Termin 2

Das Projekt 2 finden Sie im Downloadbereich.