

12.3.2 ToolBar

Die Klasse `ToolBar` (gb.form.mdi) implementiert eine *Symbolleiste* zur Aufnahme von `ToolBarButton` und `MenuBarButton`. Sie besitzt einige spezielle Eigenschaften, aber nur eine Methode und genau ein Ereignis sowie 2 Konstanten für die Festlegung der Ausrichtung einer `ToolBar`.



Abbildung 12.3.2.1: ToolBar mit `ToolBarButton` in der Gambas-IDE

12.3.2.1 Eigenschaften, Methode und Ereignis

Die beiden Eigenschaften `ToolBar.Key` und `ToolBar.Text` sind nur dann festzulegen, wenn Sie die Methode `ToolBar.Configure` einsetzen wollen. Die Methode gestattet Ihnen zur *Laufzeit* eines Programms das Design einer `ToolBar` in engen Grenzen zu ändern. Die Änderungen betreffen die Höhe der Icon auf den `ToolBar` oder `MenuBarButton` (klein, mittel oder groß), das Hinzufügen oder Entfernen von `ToolBar` oder `MenuBarButton`, das Hinzufügen oder Entfernen von Separatoren und Zwischenräumen sowie das Umgruppieren von Elementen in der `ToolBar`. Hinzufügen können Sie nur `ToolBar` oder `MenuBarButton`, die in einem Konfigurationsdialog vorher entfernt und 'geparkt' wurden. Ob dieses Feature eine gelungene Spielerei oder eine sinnvolle Ergänzung ist müssen Sie entscheiden.

12.3.2.2 Eigenschaften

ToolBar	Datentyp	Default	Beschreibung
AutoSize	Boolean	False	Gibt an, ob sich die Größe der <code>ToolBar</code> automatisch an den Inhalt anpasst
Orientation	Integer	0	Setzt die Ausrichtung der <code>ToolBar</code> oder gibt diesen Integer-Wert zurück. Entweder benutzen Sie die Zahl 0 für die horizontale Ausrichtung oder 1 für die vertikale Ausrichtung oder Sie verwenden die Konstanten <code>ToolBar.Horizontal</code> oder <code>ToolBar.Vertical</code> .
Separator	Boolean	False	Wird der Wert auf True gesetzt, erscheint unter der <code>ToolBar</code> eine dünne Linie in der Länge der <code>ToolBar</code> . Der Eigenschaftswert kann auch ausgelesen werden.

ToolTip	String	Null	Setzt den anzuzeigenden <code>ToolTip</code> -Text der <code>ToolBar</code> oder gibt diesen zurück
Text	String	Null	Setzt den Namen der <code>ToolBar</code> oder gibt diesen zurück. Der Name wird <i>intern</i> bei der <code>ToolBar</code> -Konfiguration verwendet.
Key	String	Null	Setzt einen String zur eindeutigen Identifizierung der <code>Symbolleiste</code> oder gibt ihn zurück. Dieser Schlüssel wird bei der automatischen Sicherung oder Wiederherstellung der Konfiguration der <code>Symbolleiste</code> verwendet. Nur wenn die Eigenschaft gesetzt ist, dann wird die <code>ToolBar</code> konfigurierbar sein!

Tabelle 12.3.2.2.1: Ausgewählte `ToolBar`-Eigenschaften

12.3.2.3 Methode

Die Methode `ToolBar.Configure` greift nur dann, wenn

1. ein Key-String gesetzt wurde und
2. bei den `ToolBarButton` oder `MenuBarButton` im Container `ToolBar` die `.Action`-Eigenschaft gesetzt worden ist. Diese Einstellungen sollten Sie später nicht mehr ändern.

Hinweise:

- Sie können den Key-String für eine `ToolBar` nur in der Entwicklungsumgebung (IDE) setzen.
- Gleiches gilt für die `.Action`-Eigenschaft von `ToolBarButton` oder `MenuBarButton` im Container `ToolBar`.

Die Methode `ToolBar.Configure` rufen Sie in einer Ereignisbehandlungsroutine auf oder Sie nutzen komfortabel das *Kontext-Menü* der Komponente `ToolBar`.

12.3.2.4 Ereignis

Das Ereignis `ToolBar_Configure()` wird ausgelöst, wenn die `ToolBar` konfiguriert wurde.

12.3.2.5 Beispiel

Das Beispiel demonstriert den Einsatz einer zur Programm-Laufzeit konfigurierbaren ToolBar (Symbolleiste). Die Toolbar enthält mehrere ToolButton und einen MenuButton, der jedoch *nicht* konfigurierbar ist, weil dessen .Action-Eigenschaft *nicht* gesetzt worden ist.

- Das Menü mit seinen 2 Einträgen für den MenuButton wurde mit dem Menü-Editor angelegt und konfiguriert. Die interne Programmhilfe wird über F1 aufgerufen und die Online-Hilfe über F12. Beide Funktionstasten wurden im Menü-Editor definiert.
- Die Funktionalität der beiden ToolButton *Play* und *FTP* wird nur angedeutet.



Abbildung 12.3.2.5.1: ToolBar mit ToolButton und MenuButton

Nach dem Sie in der Entwicklungsumgebung (IDE) alle Komponenten auf der Form platziert haben, müssen Sie dort im *Reiter Eigenschaften* die Eigenschaft *ToolBar.Key* und die *.Action-Eigenschaft* für die ToolButton und den MenuButton festlegen, die konfigurierbar sein sollen, weil der folgende Quelltext zwar keinen Fehler produziert, aber *wirkungslos* bleibt:

```
ToolBar1.Key = "Emma"
toolbtnOpen.Action = "A1"
toolbtnPlay.Action = "A2"
toolbtnFTP.Action = "A3"
toolbtnConfiguration.Action = "A4"
toolbtnClose.Action = "A5"
```

Minisini schreibt: *There is no code at all. You define these properties in the property sheet.*

12.3.2.6 Projekt-Quelltext

Der Quelltext wird vollständig angegeben und anschließend kommentiert:

```
[1] ' Gambas class file
[2]
[3] Public Sub Form_Open()
[4]     FMain.Center
[5]     FMain.Resizable = False
[6]     PictureBox1.Picture = Picture["fractal.jpg"]
[7]
[8]     ToolBar1.Height = 32
[9]     ToolBar1.Orientation = ToolBar1.Horizontal ' Alternative: ToolBar1.Orientation = 0
[10]    ToolBar1.Separator = False
[11]    toolbtnFTP.Toggle = True
[12]    toolbtnFTP.Value = False
[13]
[14]    mnutnHelp.Width = 72
[15]    mnutnHelp.Text = "Hilfen"
[16]    mnutnHelp.Menu = "mnuHelp"
[17]    mnutnHelp.MenuOnly = True
[18]
[19]End ' Form_Open()
[20]
[21]Public Sub mnu21Help_Click()
[22]    FHelp.Show ' → F1
[23]End ' mnuHelp_Click()
```

```

[24]
[25]Public Sub mnu22Information_Click()
[26] Desktop.Open("http://www.gambas-buch.de/dw/doku.php?id=k18:k18.6:start") ' → F12
[27]End ' mnu22Information_Click()
[28]
[29]Public Sub toolbtnOpen_Click()
[30] OpenImage()
[31]End ' toolbtnOpen_Click()
[32]
[33]Public Sub toolbtnFTP_Click()
[34] FTPStartStop()
[35]End ' toolbtnFTP_Click()
[36]
[37]Public Sub toolbtnConfiguration_Click()
[38] ToolBar1.Configure()
[39]End ' toolbtnConfiguration_Click()
[40]
[41]Public Sub toolbtnClose_Click()
[42] FMain.Close
[43]End ' toolbtnClose_Click()
[44]
[45]Public Sub Form_Close()
[46] FHelp.Close
[47] FMain.Close
[48]End ' Form_Close()
[49]
[50] '*****
[51]
[52]Private Sub OpenImage()
[53] Dialog.Title = "Importieren Sie eine Bild-Datei!"
[54] Dialog.Filter = ["*.png", "Bild-Dateien"]
[55]
[56] If Dialog.OpenFile(False) = True Then ' Multiselect=False (Standard)
[57] Message.Info("Das Öffnen der Bild-Datei wurde abgebrochen!")
[58] Return ' Cancel-Button gedrückt
[59] Else
[60] Try PictureBox1.Picture = Picture.Load(Dialog.Path)
[61] If Error Then
[62] Message.Error("Fehler beim Öffnen der Bilddatei")
[63] Else
[64] Labell.Alignment = Align.Left
[65] Labell.Text = "Datei: " & File.Name(Dialog.Path)
[66] Endif ' ERROR ?
[67] Endif ' Dialog.OpenFile(...) = TRUE ?
[68]End ' OpenImage()
[69]
[70]Private Sub FTPStartStop()
[71] If toolbtnFTP.Value = True Then
[72] toolbtnFTP.Picture = Picture["icon:/16/disconnect"]
[73] toolbtnFTP.Tooltip = "FTP-Server starten"
[74] ' FTP-Server ---> Start-Befehl ausführen
[75] Else
[76] toolbtnFTP.Picture = Picture["icon:/16/connect"]
[77] toolbtnFTP.Tooltip = "FTP-Server stoppen"
[78] ' FTP-Server ---> Stopp-Befehl ausführen
[79] Endif ' toolbtnFTP.Value = True ?
[80]End ' FTPStartStop()
[81]
[82]Public Sub ToolBar1_Configure()
[83] FMain.Background = Color.Red
[84] Wait 0.05
[85] FMain.Background = Color.ButtonBackground
[86]End ' ToolBar1_Configure()

```

Kommentare:

- In der Zeile 10 wird die Ausrichtung der ToolBar auf 'horizontal' festgelegt und dafür die Kontante *ToolBar.Horizontal* genutzt.
- Mit der Anweisung in der Zeile 11 wird festgelegt, dass unterhalb der ToolBar keine dünne Linie angezeigt wird. Statt dessen wird ein horizontaler Separator verwendet, weil der über die volle Breite der ToolBar dargestellt wird.
- Der ToolButton *toolbtnFTP* wird in der Zeile 12 als Schalter definiert und der Schalter-Zustand in

der Zeile 13 auf 'nicht eingerastet' gestellt . Die Zeilen 71 bis 81 demonstrieren die Umschaltung.

- In den Zeilen 38 bis 40 wird eine Prozedur deklariert, die den Konfigurationsdialog für die ToolBar startet.
- Die Änderung der Konfiguration der ToolBar wird in den Zeilen 83 bis 87 durch einen kurzen Farbwechsel des Programmfensters signalisiert.

12.3.2.7 Konfiguration der ToolBar

Den Konfigurationsdialog zur Konfiguration der ToolBar starten Sie entweder über den 4. ToolButton (Aufruf der Prozedur `ToolBar1.Configure()`) oder über das Kontext-Menü der ToolBar:

```
Public Sub toolbtnConfiguration_Click()
    ToolBar1.Configure()
End ' toolbtnConfiguration_Click()
```

Im Kontextmenü existieren 3 Einträge – Einrichten, Icon-Größe (3 Möglichkeiten: klein, mittel, groß) und Abbrechen. Das Ereignis `ToolBar1_Configure()` wird über das KontextMenü ausgelöst!

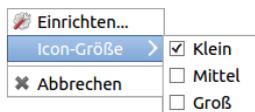


Abbildung 12.3.2.7.1: Kontext-Menü der ToolBar

Beide Varianten öffnen einen Dialog zur Konfiguration einer ToolBar. Wenn der Konfigurations-Dialog geöffnet wurde, befindet sich die ToolBar im Konfigurationsmodus. In diesem Modus wird die ToolBar deaktiviert. Sie können

- die aktuellen Inhalte der ToolBar per Drag&Drop frei in der Toolbar anordnen,
- einen ToolButton oder MenuButton aus der Symbolleiste entfernen und auf der Konfigurations-Dialog-Fläche ablegen (parken),
- ein Element aus der Konfigurations-Dialog-Fläche (Separator, Leerstelle oder erweiterte Leerstelle; optional zum Beispiel ein 'geparkter' ToolButton) in die ToolBar einfügen oder von dort auf die Konfigurations-Dialog-Fläche bewegen und ablegen sowie
- die Icon-Größe auf den Button in der ToolBar ändern (ComboBox).



Abbildung 12.3.2.7.2: Deaktivierte ToolBar und Konfigurationsdialog

Nach den Änderungen verlassen Sie den Konfigurations-Dialog über den 'Schließen'-Button. Die Symbolleiste ändert den Konfigurations-Modus und ist wieder nutzbar.

Nach einer Änderung der Geometrie des ToolBar-Konfigurationsdialog-Fensters finden Sie in der Konfigurationsdatei `~/config/gambas3/ToolBar.conf` den folgenden Eintrag:

```
[gb.form.mdi/FToolBarConfig]
Geometry=[ 1042,337,592,329,0 ]
```

oder nach einer Konfiguration der ToolBar mit diesen Einstellungen einen erweiterten Eintrag:

- Icon-Größe bleibt auf 'Klein' (Der Default-Wert 'Klein' wird nicht eingetragen)
- Einfügen von Separatoren zwischen einzelnen ToolButton
- Einfügen von Leerzeichen vor und hinter dem MenuButton

```
[gb.form.mdi/FToolBarConfig]
Geometry=[1042,337,592,329,0]

[gb.form.mdi/Toolbars/Emma]
Layout=["A1", "|", "A2", "|", "A3", "|", "A4", "-", "$mubtnHelp", "-", "A5"]
```



Abbildung 12.3.2.7.3: Konfigurierte ToolBar

Ein "Zurücksetzen" im Dialog bedeutet, dass die ToolBar-Eigenschaften, bis auf die Icon-Größe, nach einer Sicherheitsabfrage auf die Starteinstellungen zurückgesetzt werden → Abbildung 12.2.2.5.1.