

16.10 HistoryBox

Die HistoryBox ist eine TextBox, die um die Eigenschaft erweitert wurde, eine bestimmte Anzahl von Eingaben temporär zu speichern. Sie können in der HistoryBox die Pfeiltasten $\wedge \vee$ benutzen, um zum Beispiel wie in einer Konsole nach bereits eingegebenen Eingaben zu suchen und diese aufzurufen.

Die Klassen *HistoryBox*, *History* und *_HistoryOptions* wurden von Tobias Boege entwickelt.



Abbildung 16.10.1: Icon für die Komponente HistoryBox

Wie bei allen Komponenten üblich, wurde auch der HistoryBox von den Autoren ein Icon spendiert, das Sie in der IDE im Fenster Eigenschaften unter 'Form' sehen. Das Icon muss im Projektverzeichnis in einem unsichtbaren Ordner *.hidden/control* als Bild vom Typ 'png' eingefügt werden, damit es in der IDE im virtuellen Ordner 'Projekt' zu sehen ist.

Im Kapitel '21.3.5 Projekt GUI MySQL' wird der Einsatz einer HistoryBox für den dort vorgestellten Datenbank-Client ausführlich beschrieben. Sie finden dort auch die o.a. drei Klassen HistoryBox, History und _HistoryOptions sowie das erwähnte Icon im Projekt-Archiv.



Abbildung 16.10.2: GUI für das Konsolen-Programm 'mysql'

Deshalb kann hier der relevante Quelltext – bezogen auf die Komponente HistoryBox – stark verkürzt angegeben werden:

```
' Gambas class file
...
Public Sub Form_Open()
  FMain.Center
  FMain.Resizable = True
  ...
  txbHistoryEingabe.Foreground = Color.Gray
  txbHistoryEingabe.Alignment = Align.Center
  txbHistoryEingabe.Text = "*** SQL-Anweisung eingeben und mit ENTER-Taste aktivieren ***"
  txbHistoryEingabe.ReadOnly = True
  txbHistoryEingabe.History = New History(50) ' 50 Eingaben werden intern gespeichert
  ...
End ' Form_Open()

Public Sub btnProcessStart_Click()
  ...
  txbHistoryEingabe.Alignment = Align.Normal
End ' btnProcessStart

Public Sub btnTextAreaClear_Click()
  ...
  txbHistoryEingabe.SetFocus
End ' btnTextAreaClear
...
Public Sub ProcessStart()
  Dim sCommand As String

  txbHistoryEingabe.Foreground = Color.Black
  txbHistoryEingabe.Clear
  ...
  txbHistoryEingabe.ReadOnly = False
  txbHistoryEingabe.SetFocus
End ' ProcessStart()
```

```
...  
Public Sub txbHistoryEingabe_Activate()  
    If txbHistoryEingabe.Text = "" Then Return  
    txbHistoryEingabe.Text = Trim(txbHistoryEingabe.Text)  
    WriteToMyProcess(txbHistoryEingabe.Text)  
    ' txbHistoryEingabe.Clear ---> Das erledigt die History-Box selbst!  
End ' txbHistoryEingabe_Activate()  
...  
Public Sub Form_Close()  
...  
End ' Form_Close()
```

→