

17.7.5 GridView – Daten-Import aus einer Datenbank-Tabelle

Dieses Projekt demonstriert die Anzeige von Daten aus einer Datenbank-Tabelle in einer GridView. Beim Programmstart ist die Anzahl der Zeilen und Spalten in der Gitteransicht nicht bekannt, sondern abhängig von der SQL-Anfrage. Von großem Vorteil ist die Struktur des Resultates aus Datensätzen (Zeilen) und Feldern (Spalten), das vom Datenbank-Server als Matrix zurückgeliefert wird. Da somit eine Konvertierung oder Zwischenspeicherung der gelieferten Daten entfällt, können diese sofort in die GridView eingefügt und angezeigt werden. Im Projekt wird eine SQLite3-Datenbank-Tabelle eingesetzt:

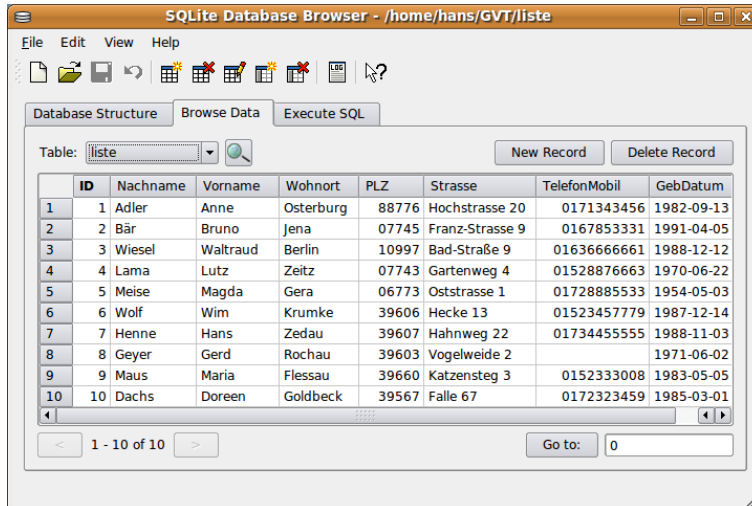


Abbildung 17.7.5.1: Anzeige der DB-Daten im SQLite Database Browser

Mit diesem Programm-Quelltext können Sie Daten aus einer Datenbank-Tabelle in einer GridView anzeigen:

```
[1] ' Gambas class file
[2]
[3] PUBLIC rDBResult AS Result
[4] PUBLIC cDBVerbindung AS NEW Connection
[5]
[6] PUBLIC SUB Form_Open()
[7]     FMain.Center
[8]     FMain.Border = 1
[9]     GridView.Rows.Count = 0 ' Beim Programmstart wird KEIN Gitter angezeigt
[10] END ' Form_Open
[11]
[12] PUBLIC SUB btnDBDatenImGitterAnzeigen_Click()
[13]     GridView.Clear
[14]     GridView.Rows.Count = 0
[15]     TRY DBDatenAuslesen()
[16]     IF ERROR THEN
[17]         Message.Error("Es konnten keine DB-Daten ausgelesen werden!")
[18]         RETURN
[19]     ENDIF ' ERROR ?
[20]
[21]     IF rDBResult.Count = 0 THEN
[22]         Message.Info("Die Anzahl der selektierten Datensätze ist Null!")
[23]         RETURN
[24]     ELSE
[25]         SetGridViewProperty()
[26]         GridView.Rows.Count = rDBResult.Count ' Jetzt wird das Gitter gezeichnet!
[27]         GridView.SetFocus
[28]         ' Sprung in die letzte Datenzeile ----> GridView.MoveTo(rDBResult.Count - 1, 0)
[29]         ' Sprung in die erste Datenzeile ----> GridView.MoveTo(0, 0)
[30]         GridView.MoveTo(0, 0)
[31]     ENDIF ' rDBResult.Count = 0 ?
[32] END ' DBDatenImGitterAnzeigen
[33]
[34] PUBLIC SUB SetGridViewProperty()
[35]     DIM iDatensatzNummer, iSpaltenNummer AS Integer
[36]
[37]     GridView.Header = 1
[38]     GridView.Columns.Count = rDBResult.Fields.Count
[39]     GridView.Columns[0].Width = 25
```

```
[40] GridView.Columns[0].Resizable = FALSE ' Die Feldbreite ID ist fix
[41] ' GridView-Spalten-Bezeichner ermitteln und eintragen
[42] FOR iSpaltenNummer = 0 TO rDBResult.Fields.Count - 1
[43]     GridView.Columns[iSpaltenNummer].Title = rDBResult.Fields[iSpaltenNummer].Name
[44] NEXT ' Feldname
[45] END ' SetGridViewProperty
[46]
[47] PUBLIC SUB DBDatenAuslesen()
[48]     DIM sSQL_Anweisung AS String
[49]
[50]     cDBVerbindung.Close()
[51]     cDBVerbindung.Type = "sqlite3" ' Der Typ muss klein geschrieben werden!
[52]     cDBVerbindung.Host = Application.Path & / "SQL/GVT" ' User.Home & / "GVT"
[53]     cDBVerbindung.Name = "Liste" ' Das ist der Datenbank-Name
[54]     cDBVerbindung.User = "" ' ---> Nur bei MySQL und PostgreSQL erforderlich
[55]     cDBVerbindung.Password = "" ' ---> Nur bei MySQL und PostgreSQL erforderlich
[56]     cDBVerbindung.Port = "" ' ---> Nur bei MySQL und PostgreSQL erforderlich
[57]
[58] ' Versuch, eine DB-Verbindung herzustellen
[59] TRY cDBVerbindung.Open()
[60] IF ERROR THEN
[61]     Message.Error("Eine DB-Verbindung zum DB-Server konnte NICHT hergestellt werden!")
[62] ENDIF
[63]
[64]     sSQL_Anweisung = "SELECT * FROM " & "liste" & " ORDER BY Nachname"
[65]     rDBResult = cDBVerbindung.Exec(sSQL_Anweisung)
[66]
[67] END ' DBDatenAuslesen
[68]
[69] PUBLIC SUB GridView_Data(Row AS Integer, Column AS Integer)
[70]     rDBResult.MoveTo(Row)
[71]     GridView.Data.Text = rDBResult[Column]
[72]     IF row MOD 2 = 0 THEN GridView.Data.Background = Color.RGB(224, 224, 224)
[73] END ' grdWerteTabelle_Data
[74]
[75] PUBLIC SUB Form_Close()
[76]     TRY cDBVerbindung.Close()
[77]     IF ERROR THEN Message.Error("Fehler bei Trennen der DB-Verbindung!")
[78] END ' Form_Close
[79]
[80] PUBLIC SUB btnProgrammEnde_Click()
[81]     FMain.Close
[82] END ' Programm-Ende
```

Der Quelltext birgt keine Überraschungen bis auf auf die Prozedur GridView_Data(..) in den Zeilen 69 bis 73, mit deren Hilfe die Daten 'en bloc' in die GridView eingefügt und angezeigt werden können:

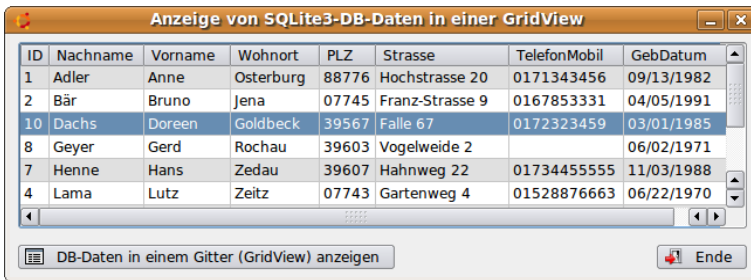


Abbildung 17.7.5.2: Anzeige von DB-Daten in einer GridView

Der Unterschied in den beiden Abbildungen resultiert aus dem letzten Teil der SQL-Anweisung, in der eine Sortierung nach den Nachnamen gefordert wird. In der Abbildung 17.7.5.1 werden die Datensätze so angezeigt, wie sie in die Datenbank-Tabelle eingefügt wurden.