

### 22.5.1 Klasse DataBrowser (gb.db.form)

Die Klasse *DataBrowser* präsentiert ein DB-Gridview, die den Inhalt eines Ergebnis-Objektes (Datentyp Result) anzeigt. Es kann eine Navigationsleiste zugeschaltet werden, um durch die Datensätze zu navigieren. Über die Eigenschaft 'Columns' können Sie festlegen, welche DB-Felder angezeigt und bearbeitet werden können. Die DB-Daten werden über das Steuerelement 'Datasource' bereitgestellt.

#### 22.5.1.1 Eigenschaften

Die Klasse *DataBrowser* verfügt über die folgenden Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
CanCreate	Boolean	Für den Wert True besitzt der DataBrowser einen CreateRecord-Button. Sie können den Wert der Eigenschaft auch auslesen. Es kann ein neuer Datensatz erzeugt werden. Die Voreinstellung ist True.
CanDelete	Boolean	Für den Wert True besitzt der DataBrowser einen Lösch-Button. Sie können den Wert der Eigenschaft auch auslesen. Es kann ein ausgewählter Datensatz gelöscht werden. Die Voreinstellung ist True.
Columns	String[ ]	Setzt die Felder, die im DataBrowser in den Spalten angezeigt werden. Sie können den Wert der Eigenschaft auch auslesen. Ist die Eigenschaft nicht gesetzt, dann fehlt diese Beschränkung und es werden automatisch <i>alle</i> Felder angezeigt.
Control	Boolean	Ist der Wert True, werden <i>alle</i> Icon des Steuerelements angezeigt wie New, Save, Delete, Refresh, MoveNext, MovePrevious, MoveFirst und MoveLast. Sie können den Wert der Eigenschaft auch auslesen. Die Voreinstellung ist True.
Data	_GridView_Data	Verwenden Sie diese Eigenschaft, um den Inhalt einer Zelle während des Data-Ereignisses zu speichern (Event Data ( Row As Integer, Column As Integer, Value As Variant )).
Editable	Boolean	Ist der Wert True, können die Inhalte aller Zellen in der TableView editiert werden. Sie können den Wert der Eigenschaft auch auslesen.
Grid	Boolean	Ist der Wert True, wird ein Gitter angezeigt. Sie können den Wert der Eigenschaft auch auslesen.
Header	Boolean	Ist der Wert True, wird eine Kopfzeile im Gitter angezeigt. Sie können den Wert der Eigenschaft auch auslesen.
Labels	String[ ]	Ersetzt die Feld-Namen in der Kopfzeile im Gitter durch den Inhalt des String-Arrays in der vorgegebenen Reihenfolge. Sie können den Wert der Eigenschaft auch auslesen.
Orientation	Integer	Gibt an, ob die Icon-Liste oben, rechts, unten oder links im DataBrowser angezeigt werden. Verwenden Sie zur Angabe der Orientierung die Konstanten Align.Top (19), Align.Bottom (35), Align.Right (2) oder Align.Left (1). Sie können den Wert der Eigenschaft auch auslesen. Die Voreinstellung ist 1.
View	TableView	Gibt die für die Anzeige der DataBrowser-Inhalte verwendete TableView zurück.

Tabelle 22.5.1.1.1 : Eigenschaften der Klasse DataBrowser

#### 22.5.1.2 Methoden

Die Klasse *DataBrowser* verfügt nur über diese spezielle Methode *DataBrowser.Update ( )*. Sie lädt den (aktuellen) Daten-Inhalt erneut in den DataBrowser.

#### 22.5.2 Ereignisse

Die Klasse *DataBrowser* hat nur ein Ereignis *Data ( Row As Integer, Column As Integer, Value As Variant )*, das für den DataBrowser in Bezug auf die Daten von Belang ist:

- Row ist der Zeilenindex.
- Column ist der Spaltenindex.

- Value ist der Wert, der von der Datenbank erhalten wurde und in der Zelle angezeigt wird.

Der Ereignis wird ausgelöst, wenn der DataBrowser eine bestimmte Zelle anzeigen muss. Verwenden Sie die Eigenschaft Data, um den Inhalt einer Zelle zu definieren.

### 22.5.2.1 Beispiele

Die Tabelle 'contacts' in der SQLite3-Datenbank 'contacts.sqlite' wurde mit dieser SQL-Anweisung erzeugt und anschließend mit Daten gefüllt:

```
hans@mint-183 ~ $ sqlite3 contacts.sqlite
SQLite version 3.11.0 2016-02-15 17:29:24
Enter ".help" for usage hints.
sqlite> CREATE TABLE IF NOT EXISTS contacts
...> (
...> id INTEGER PRIMARY KEY AUTOINCREMENT,
...> vorname TEXT,
...> nachname TEXT,
...> wohnort TEXT,
...> plz TEXT,
...> strasse TEXT,
...> email TEXT
...> );
sqlite> .databases
seq name file
-----
0 main /home/hans/contacts.sqlite
sqlite> .tables
contacts
sqlite> .quit
hans@mint-183 ~ $
```

Die Anzeige aller Datensätze – die von einem gesetzten Daten-Filter nicht ausgefiltert werden – übernimmt das Steuer-Element DataBrowser. Für diese Konfiguration ergibt sich das folgende Bild:

```
Private Sub SetDBBrowserProperties()

    DataBrowser1.Columns = ["vorname", "nachname", "plz", "wohnort", "strasse", "email"]
    DataBrowser1.Labels = ["Vorname", "Nachname", "PLZ", "Wohnort", "Strasse", "EMail-Adresse"]

    DataBrowser1.Control = True ' Default: True
    DataBrowser1.CanCreate = True ' Default: True
    DataBrowser1.CanDelete = False ' Default: True
    DataBrowser1.Editable = False

    DataBrowser1.View.Clear()

    DataBrowser1.View.Columns[0].Width = 90
    DataBrowser1.View.Columns[1].Width = 90
    DataBrowser1.View.Columns[2].Width = 60
    DataBrowser1.View.Columns[3].Width = 120
    DataBrowser1.View.Columns[4].Width = 125

    DataBrowser1.View.MoveTo(0, 0) ' Required

End
```

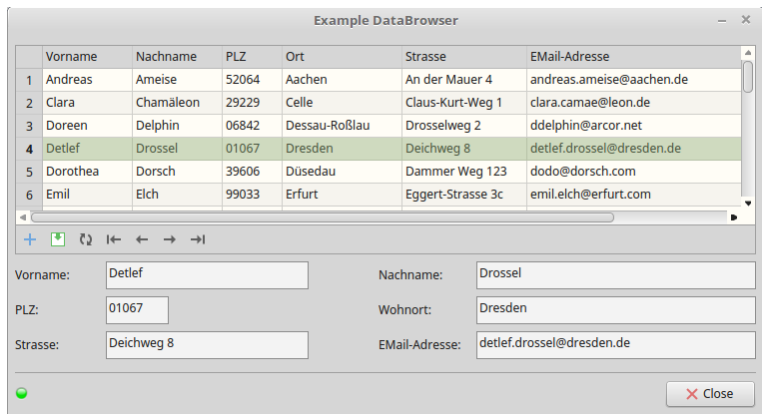


Abbildung 22.5.2.1.1: Anzeige 1

Die Konfiguration wurde so geändert:

```
Private Sub SetDBBrowserProperties()

    DataBrowser1.Columns = ["id", "vorname", "nachname", "plz", "wohntort", "strasse", "email"]
    DataBrowser1.Labels = ["ID", "Vorname", "Nachname", "PLZ", "Wohnort", "Strasse", "EMail-Adresse"]

    DataBrowser1.Control = True ' Default: True
    ' DataBrowser1.CanCreate = True ' Default: True
    ' DataBrowser1.CanDelete = True ' Default: True
    DataBrowser1.Editable = True

    DataBrowser1.View.Clear()

    DataBrowser1.View.Columns[0].Width = 30
    DataBrowser1.View.Columns[1].Width = 90
    DataBrowser1.View.Columns[2].Width = 90
    DataBrowser1.View.Columns[3].Width = 45
    DataBrowser1.View.Columns[4].Width = 120
    DataBrowser1.View.Columns[5].Width = 125
    DataBrowser1.View.Columns[6].Width = 1

    DataBrowser1.View.MoveTo(0, 0) ' Required

End
```

### Kommentar

- Es werden mit der geänderten Konfiguration jetzt alle Felder im *DataBrowser* angezeigt.
- Alle Icons in der Navigationsleiste sind sichtbar.
- Die einzelnen Feld-Inhalte können in der Tabelle selbst geändert werden (Edit-Modus).
- Aktuell wird der Datensatz mit dem Feldwert id = 7 geändert.
- Für das letzte Feld wird eine Feldbreite angegeben – hier mit dem Wert 1. Damit wird das letzte Feld nur so breit, wie es der verbleibende Platz zulässt. Ein horizontaler Scroll-Balken entfällt.

### Hinweis

- Die Felder können mit einem Klick auf den (Anzeige-)Feldnamen sortiert werden. Dabei gilt: Erster Klick: absteigende Sortierung (▼), zweiter Klick: aufsteigende Sortierung (▲) und dritter Klick: ohne Sortierung – es wird kein Symbol angezeigt.

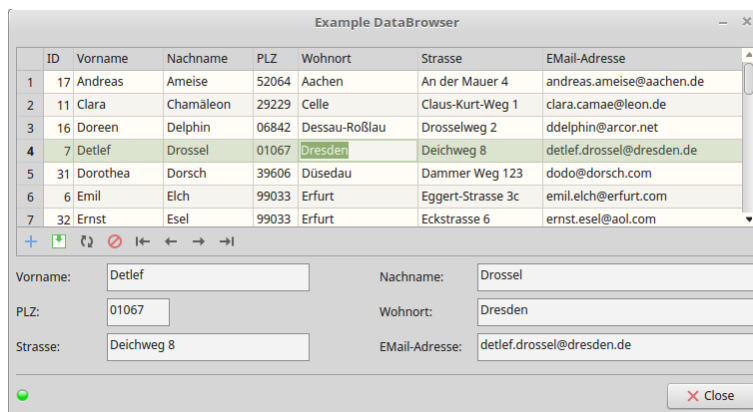


Abbildung 22.5.2.1.2: Anzeige 2

Das Steuerelement 'DataSource' stellt den Steuerelementen im Container alle Daten der ausgewählten Datenbank zur Verfügung, zu der verbunden wurde. Welche DB-Daten tatsächlich zur Anzeige zur Verfügung stehen, können Sie über die Eigenschaften *Table* und *Filter* auswählen:

```
DataSource1.Connection = DBCS.DBConnection

DataSource1.Table = "contacts"

'--- Filtering wird ausgeschaltet
DataSource1.Filter = "id > 4 AND wohntort <> 'Berlin' AND wohntort NOT LIKE 'G%'"
```

Ohne zusätzlichen Filter ergibt sich zusammen mit der zweiten Konfiguration diese Anzeige 3:

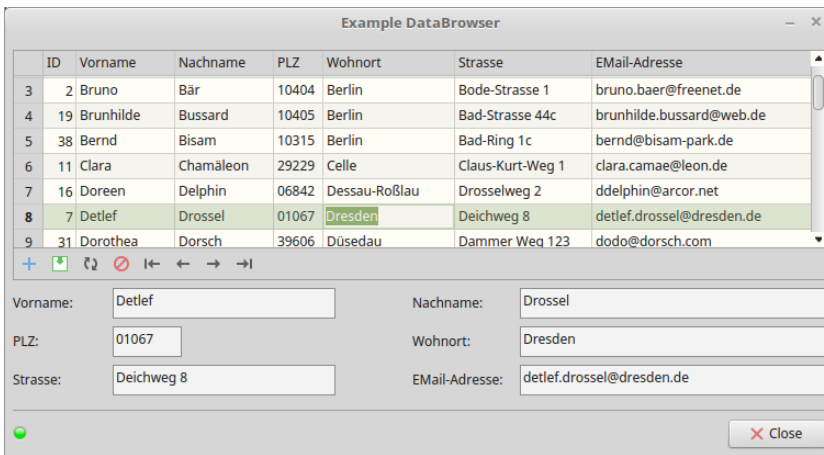


Abbildung 22.5.2.1.3: Anzeige 3

So ermitteln Sie die markierte Zeile im *DataBrowser*:

```
selectedRow = DataBrowser.View.Row
```