

Lazarus und die dBASE IV Datenbank

Werner Gatzweiler
Rheinertstraße 7
D-52525 Heinsberg

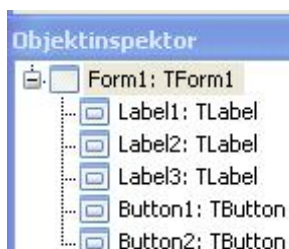
Mit diesem Beitrag möchte ich dazu beitragen, dass es Anfängern wie mir gelingt Datenbankanwendungen mit Lazarus zu erzeugen.

Für dBASE IV habe ich mich entschieden, da Lazarus alles Notwendige mit der TDBF mitbringt und viele andere OpenSource-Programme dBASE IV unterstützen.

Die Informationen zu dieser Anleitung habe ich zum Teil den Internet, dem deutschen und englischen Lazarusforum entnommen. Letztendlich hat auch das deutsche Handbuch „Lazarus, arbeiten mit IDE und Klassenbibliothek“ aus dem Verlag Computer & Literatur seinen Weg in meinen Bücherschrank gefunden.

Starten wir also Lazarus, fangen wir an.

Mit dem Start von **Lazarus – Datei – Neu – Anwendung** wird sofort eine Unit1 angelegt. Zu dieser Unit1 gehört auch automatisch die Form1, die ich wie folgt aufgezogen habe.



Der Objectinspector zeigt, dass die Form1 aus einfachen Labeln und Button besteht.

Button1 und Button2 haben in der Eigenschaftstabelle unter Caption die Aufschrift „Start“ und „Ende“ bekommen.

Wir legen nun mit **Datei-Neues Formular** eine neue Unit2 mit dem dazugehörigen Formular2 an. Nachdem dies getan ist, kehren wir wieder zur Unit1 zurück um diese zu vervollständigen.

```
unit Unit1;
{$mode objfpc}{$H+}
interface
uses
  Classes, SysUtils, db, dbf, FileUtil, Forms, Controls, Graphics, Dialogs,
  StdCtrls, unit2;
type
  { TForm1 }
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;
var
  Form1: TForm1;
implementation
{$R *.lfm}
{ TForm1 }
procedure TForm1.Button2Click(Sender: TObject);
begin
  // Abfrage einbauen
  //ShowMessage('soll das Programm wirklich beendet werden?');
  //obige Abfrage ist zu einfach!
  Case QuestionDlg('Abschlussinformation','Soll das Programm wirklich beendet werden?',
    mtCustom,[mrYes,'Ja',mrNo,'Nein'],0) of
    mrYes: close;
  end;
end;
```

Wir tragen hier den Namen der Unit2 ein, damit Unit1 diese kennt.

Mit einem Klick auf den Button2 mit der Caption „Ende“ erreichen wir diese Prozedur. Mit einer Abfrage stellen wir sicher, dass wir unser Programm nicht zu früh beenden.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
    Form2.show;
```

```
    Form2.setFocus;
```

```
end;
```

```
end.
```

Ein Klick auf den Button1 mit der Caption „Start“ führt uns zu dieser Procedur. In dieser machen wir die Form2 sichtbar und setzen den Focus auf Form2.

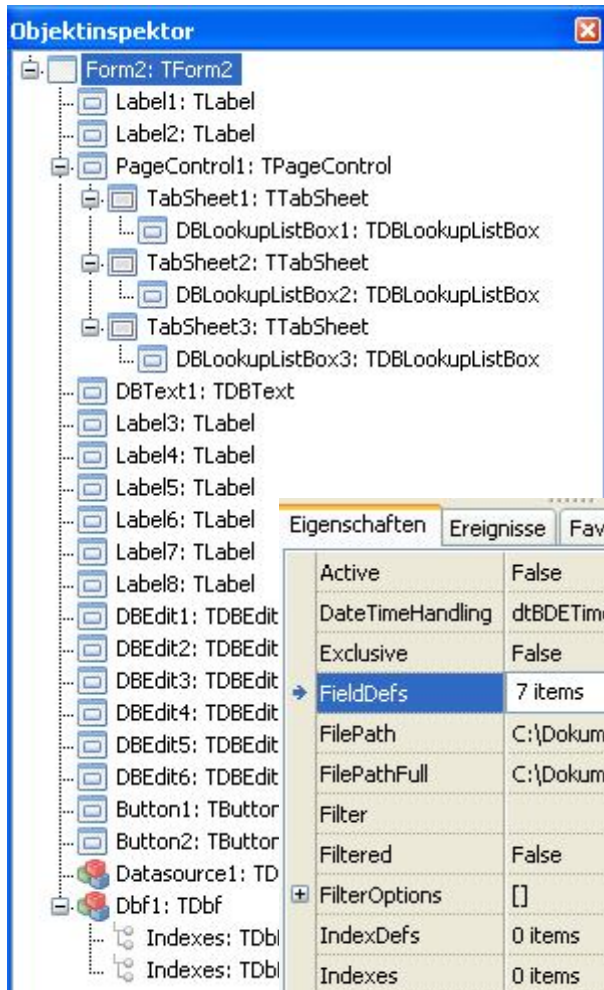
Dies ist das Ende der Unit1.

So, kommen wir jetzt zu unser Form2 und damit zur Unit2.

Auf der Unit2 werden wir unsere ersten Datenbankanweisungen und Controls platzieren.

The screenshot shows the Lazarus IDE's design view for a form titled 'Form2'. The form is set against a dotted grid background. In the top-left corner, a 'Datasource1' component is connected to a 'Dbf1' dataset. Below this, a list box displays three entries: 'Egon', 'Hans', and 'Luise'. To the right of the list box, the 'Record-Nr.' is set to '1'. Below the record number, there are several text input fields: 'Titel' (containing 'Dr.'), 'Vorname' (containing 'Egon'), 'Nachname' (containing 'Dingsda'), 'Strasze' (containing 'Erkenweg 12'), and 'PLZ u. Ort' (containing '28425' and 'Heiligenhausen'). At the bottom of the form, there are two buttons: 'Neuaufnahme' and 'zurück'.

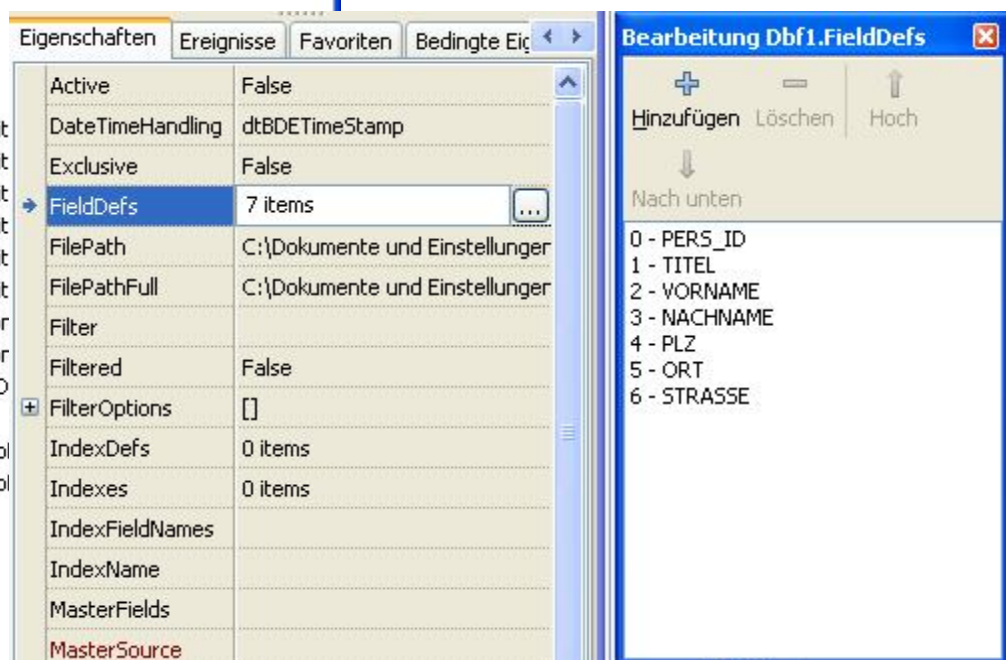
Die Form2 sieht schon etwas komplexer aus und der Objektinspektor gibt uns Auskunft darüber.



Der Objektinspektor zeigt uns die Elemente, die auf der Form2 abgelegt sind.

Als erstes legen wir die TDBF aus dem Reiter Data Access auf Formular, dann holen wir uns Tdatasource ebenfalls aufs Formular. Die Reihenfolge ist eigentlich egal. Dann speichern wir vielleicht einmal unsere Arbeit, denn man weiß ja nie?!

Als nächstes klicken wir auf das Icon TDBF und erzeugen uns die Datenbankfelder, indem wir auf dem Eigenschaftsfeld **FieldDefs** anwählen und dann



die entsprechenden Felder einfügen, die entsprechenden Datentypen und Feldlängen setzen.

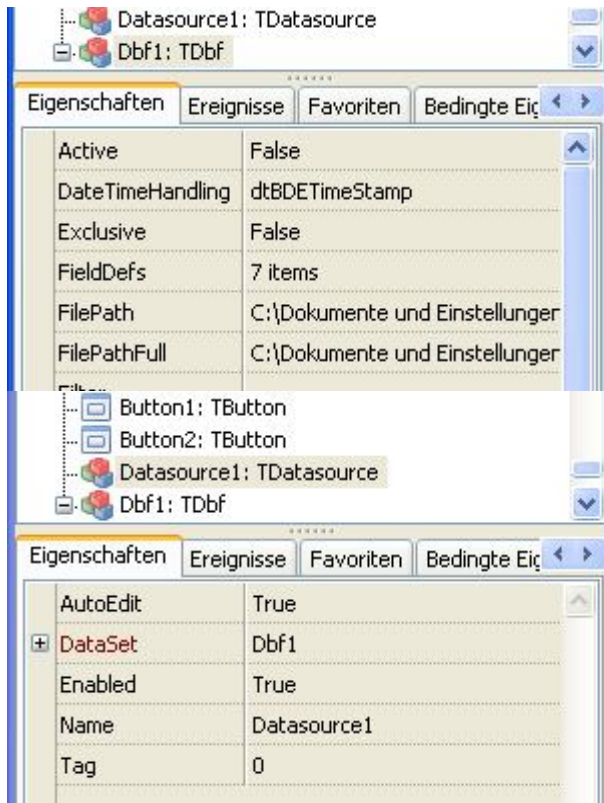
Ist dies getan, geben wir unserer dBASE-Datei einen Namen. Das geschieht ebenfalls im



Eigenschaftsmenü. Hier trägt der TableName die Bezeichnung „personal.dbf“. Des weiteren setzen wir TableLevel auf „4“, StoreDefs auf „true“ und OpenMode auf „omAutoCreate“.

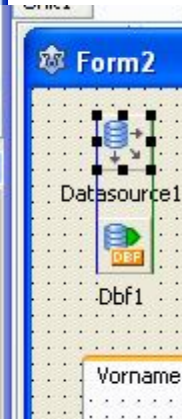
Zwischendurch das Speichern nicht vergessen.

Wenn das getan ist, sagen wir noch, wo die Datei zu finden ist.



Das sagen wir ebenfalls in der Eigenschaftsmaske und zwar unter FilePath.

Jetzt folgt noch die Anbindung an Datasource.

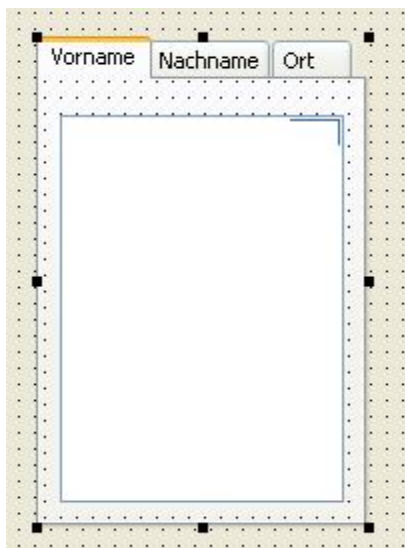


Ein Klick auf Datasource und wir tragen im Eigenschaftsmenü unter DataSet unsere **Dbf1** ein.

Jetzt noch einmal Dbf1 anklicken und Active auf „True“ setzen.



Jetzt kommen die anderen Elemente der Form2.



Vom Reiter „Common Controls“ holt man sich das Element „TpageControl“ und fügt diesem die Seiten TtabSheet mit den Überschriften „Vorname, Nachname und Ort“ hinzu.

Dann klickt man auf die erste Seite, hier z.B. „Vorname“ und platziert da drauf das Element „TDBLookupListBox“ vom Reiter „Data Controls“.

Das macht man so mit den Seiten „Nachname und Ort“ gleichfalls.

Record-Nr.:	1
Titel:	Dr.
Vorname:	Egon
Nachname:	Dingsda
Strasze:	Erkenweg 12
PLZ u. Ort:	28425 Heiligenhausen

Die Bezeichnungen „Record-Nr.:“, „Titel:“, „Vorname“ usw. stehen in der Caption von einfachen „Tlabel“.

Die dargestellte Record-Nr. „1“ steht auf einem TDBText-Feld, hingegen sind die anderen Ausgaben auf TDBEdits platziert.

Die Button „zurück“ weisen wieder auf Unit1 und „Neuaufnahme“ zu der noch zu erstellenden Unit3.

Also, wie gehabt **Datei-Neues Formular** erzeugt die Unit3 mit der Form3.

Ist dies getan wieder zurück zur Unit2 und dann dort folgendes nachtragen, wie wir es schon von der Unit1 kennen.

uses

Classes, SysUtils, db, dbf, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls, ComCtrls, DbCtrls, **Unit3**;

```
procedure TForm2.Button2Click(Sender: TObject);
begin
  close;
end;
```

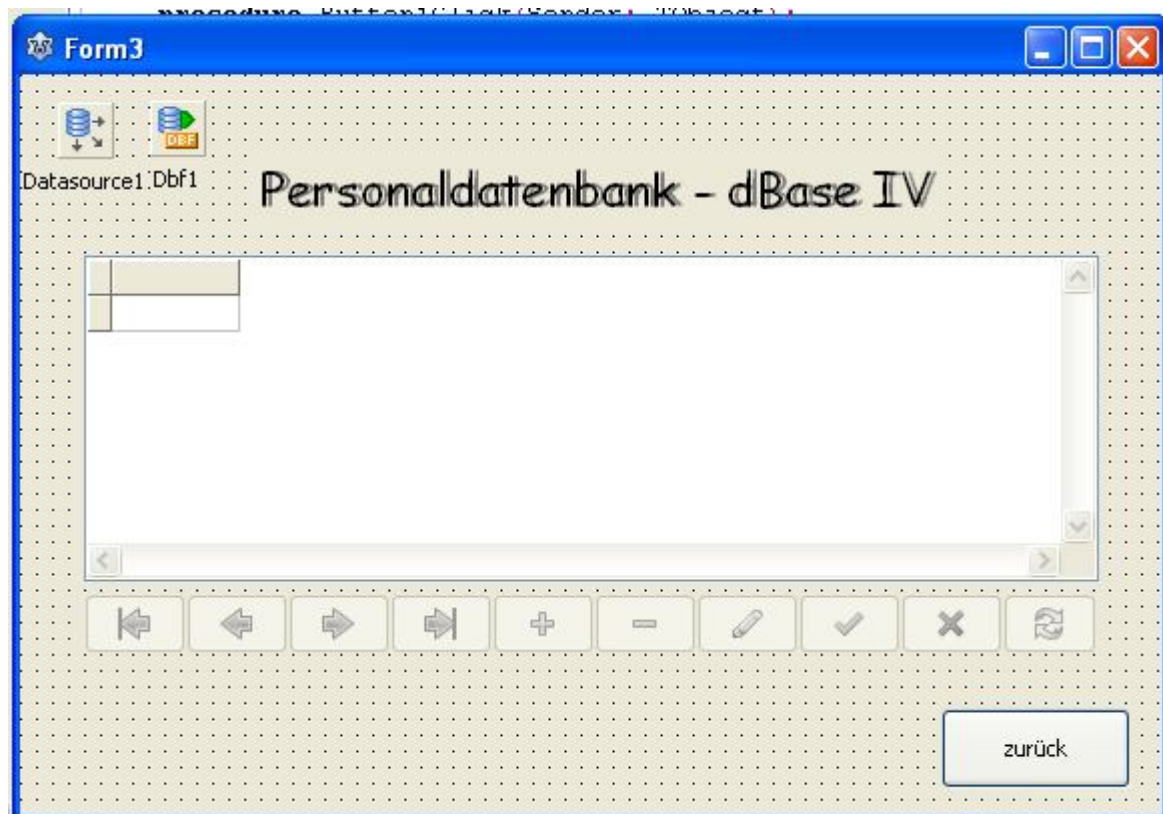
Diese Prozedur führt uns wieder zur Form1.

```
procedure TForm2.Button1Click(Sender: TObject);
begin
  Form3.show;
  Form3.setFocus;
end;
```

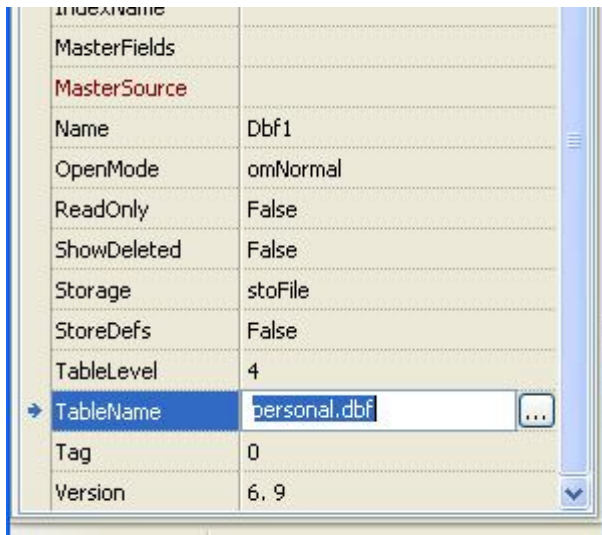
Diese Prozedur führt uns zur Form3 und zeigt sie an.

Auf zur Unit3.

Die Form3 der Unit3 gestalten wir uns wie folgt.



Auch hier ziehen wir uns die elemente Datasource und Dbf1 auf die Form. Zusätzlich holen wir uns noch TDBGrid und den TDBnavigator hinzu. Der Button „zurück“ ist glaube ich jetzt schon oft genug beschrieben.



Wir klicken Dbf1 an und suchen uns unsere „personal.dbf“. Wenn wir diese angewählt haben, dann trägt sie sich selbst und die entsprechenden Pfade ein.

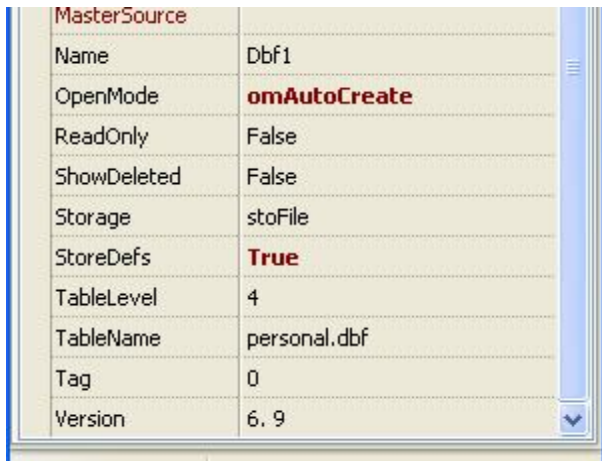
Dann ein Klick auf Datasource1 und wir tragen da als Dataset ebenfalls „Dbf1“ ein.

Nicht vergessen, immer wieder **SICHERN!**

Wenn das alles getan ist, wieder „Dbf1“ klicken und Active auf „True“ setzen.

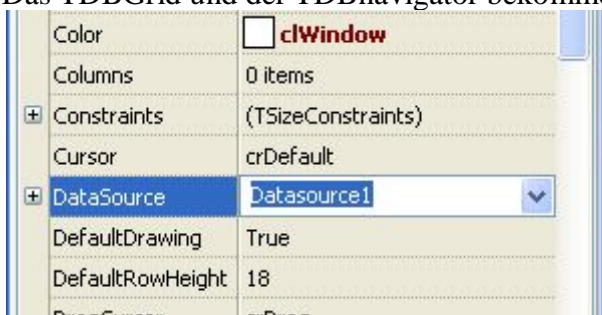
Damit ist die dBASE-Datei eingebunden.

Als letztes nochmals „Dbf1“ klicken und OpenMode und StoreDefs anpassen.



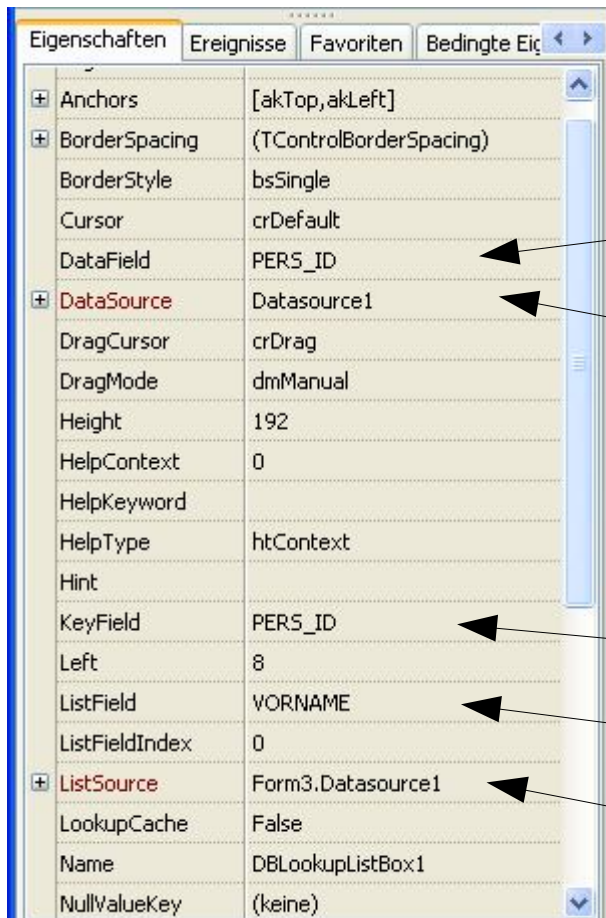
So sollte dies aussehen.

Das TDBGrid und der TDBnavigator bekommen ebenfalls die Zuweisung:



Damit auch diese elemente wissen, woher ihre Daten kommen.

Jetzt müssen wir aber noch einmal zur Unit2 zurück.



Den **DBLookupListBoxen** müssen wir noch sagen, wie sie sich verhalten sollen.

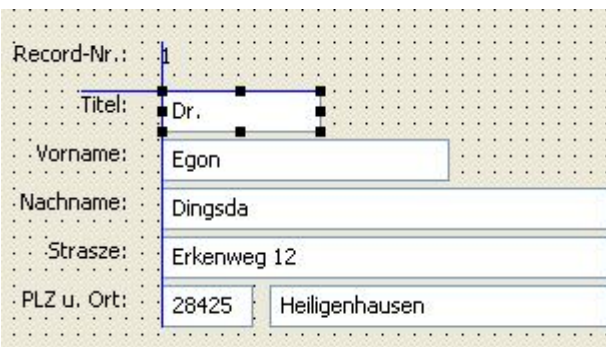
Verbindungsglied ist das DataField **PERS_ID**,

Die Datenquelle steht in **Datasource1**.

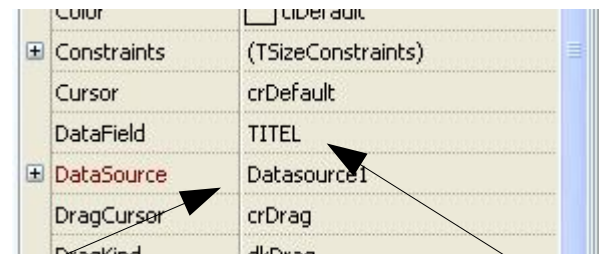
Verbindungsglied ist das DataField **PERS_ID**,

das Listenfeld ist einmal der **Vorname**, der **Nachname** und einmal der **Ort**

Die Listenquelle ist **Form3.Datasource1** der Unit3.



Die anderen Anzeigeelemente (TDBEdits) beziehen ihre Werte aus:



Datasource1 und dem zugehörigem DataField.

Der Pfiff des Ganzen ist, das ein Klick in die TDBLoopupListBox diese focusiert, dann ein Klick



auf den Eintrag, der richtige Datensatz mit Inhalt angezeigt wird.

Ermöglicht wird dies durch folgende Prozedur.

```
procedure TForm2.DBLookupListBox1Click(Sender: TObject);  
begin  
  begin  
    if not Datasource1.DataSet.Locate('PERS_ID',DBText1.caption,[loCaseInsensitive, loPartialKey])  
    then  
      MessageDlg('Suchrecord nicht gefunden.', mtInformation, [mbOK], 0);  
      //edSearch.Color:=clRed;  
    end;  
  
  end;
```

Ich hoffe, das man mit dieser Anleitung arbeiten kann.

Für Anregungen habe ich immer ein offenes Ohr, ins Besondere deshalb, da ich mich noch als Anfänger in der Sache LAZARUS sehe.

Und Danke an alle Forenmitglieder, ohne deren Anregungen ich das nicht geschafft hätte.

Mit Gruß

Werner Gatzweiler

Im Anhang noch die Einzelnen Units.

```
unit Unit1;

{$mode objfpc}{$H+}

interface

uses
  Classes, SysUtils, db, dbf, FileUtil, Forms, Controls, Graphics, Dialogs,
  StdCtrls, unit2;

type
  { TForm1 }

  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.lfm}

{ TForm1 }

procedure TForm1.Button2Click(Sender: TObject);
begin
  // Abfrage einbauen
  //ShowMessage('soll das Programm wirklich beendet werden?');
  //obige Abfrage ist zu einfach!

  Case QuestionDlg('Abschlussinformation','Soll das Programm wirklich beendet werden?',
    mtCustom,[mrYes,'Ja',mrNo,'Nein'],0) of
    mrYes: close;
  end;
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    Form2.show;
    Form2.setFocus;
end;

end.
```

```
unit Unit2;

{$mode objfpc}{$H+}

interface

uses
  Classes, SysUtils, db, dbf, FileUtil, Forms, Controls, Graphics, Dialogs,
  StdCtrls, ComCtrls, DbCtrls, Unit3;

type

  { TForm2 }

  TForm2 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Datasource1: TDataSource;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    DBEdit3: TDBEdit;
    DBEdit4: TDBEdit;
    DBEdit5: TDBEdit;
    DBEdit6: TDBEdit;
    Dbf1: TDbf;
    DBLookupListBox1: TDBLookupListBox;
    DBLookupListBox2: TDBLookupListBox;
    DBLookupListBox3: TDBLookupListBox;
    DBText1: TDBText;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    TabSheet3: TTabSheet;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure DBLookupListBox1Click(Sender: TObject);
    procedure DBLookupListBox2Click(Sender: TObject);
    procedure DBLookupListBox3Click(Sender: TObject);
  private
    { private declarations }
  end;
```



```
public
{ public declarations }
end;

var
  Form2: TForm2;

implementation

{$R *.lfm}

{ TForm2 }

procedure TForm2.Button2Click(Sender: TObject);
begin
  close;
end;

procedure TForm2.DBLookupListBox1Click(Sender: TObject);
begin
  begin
    if not Datasource1.DataSet.Locate('PERS_ID',DBText1.caption,[loCaseInsensitive, loPartialKey])
    then
      MessageDlg('Suchrecord nicht gefunden.', mtInformation, [mbOK], 0);
      //edSearch.Color:=clRed;
    end;
  end;

end;

procedure TForm2.DBLookupListBox2Click(Sender: TObject);
begin
  begin
    if not Datasource1.DataSet.Locate('PERS_ID',DBText1.caption,[loCaseInsensitive, loPartialKey])
    then
      MessageDlg('Suchrecord nicht gefunden.', mtInformation, [mbOK], 0);
      //edSearch.Color:=clRed;
    end;
  end;

end;

procedure TForm2.DBLookupListBox3Click(Sender: TObject);
begin
  begin
    if not Datasource1.DataSet.Locate('PERS_ID',DBText1.caption,[loCaseInsensitive, loPartialKey])
    then
      MessageDlg('Suchrecord nicht gefunden.', mtInformation, [mbOK], 0);
      //edSearch.Color:=clRed;
    end;
  end;

end;
```

end;

```
procedure TForm2.Button1Click(Sender: TObject);  
begin  
    Form3.show;  
    Form3.setFocus;  
end;
```

end.

```
unit Unit3;

{$mode objfpc}{$H+}

interface

uses
  Classes, SysUtils, db, dbf, FileUtil, Forms, Controls, Graphics, Dialogs,
  StdCtrls, DBGrids, DbCtrls;

type
  { TForm3 }

  TForm3 = class(TForm)
    Button1: TButton;
    Datasource1: TDataSource;
    Dbf1: TDbf;
    DBGrid1: TDBGrid;
    DBNavigator1: TDBNavigator;
    Label1: TLabel;
    Label2: TLabel;
    procedure Button1Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;

var
  Form3: TForm3;

implementation

{$R *.lfm}

{ TForm3 }

procedure TForm3.Button1Click(Sender: TObject);
begin
  close;
end;

end.
```