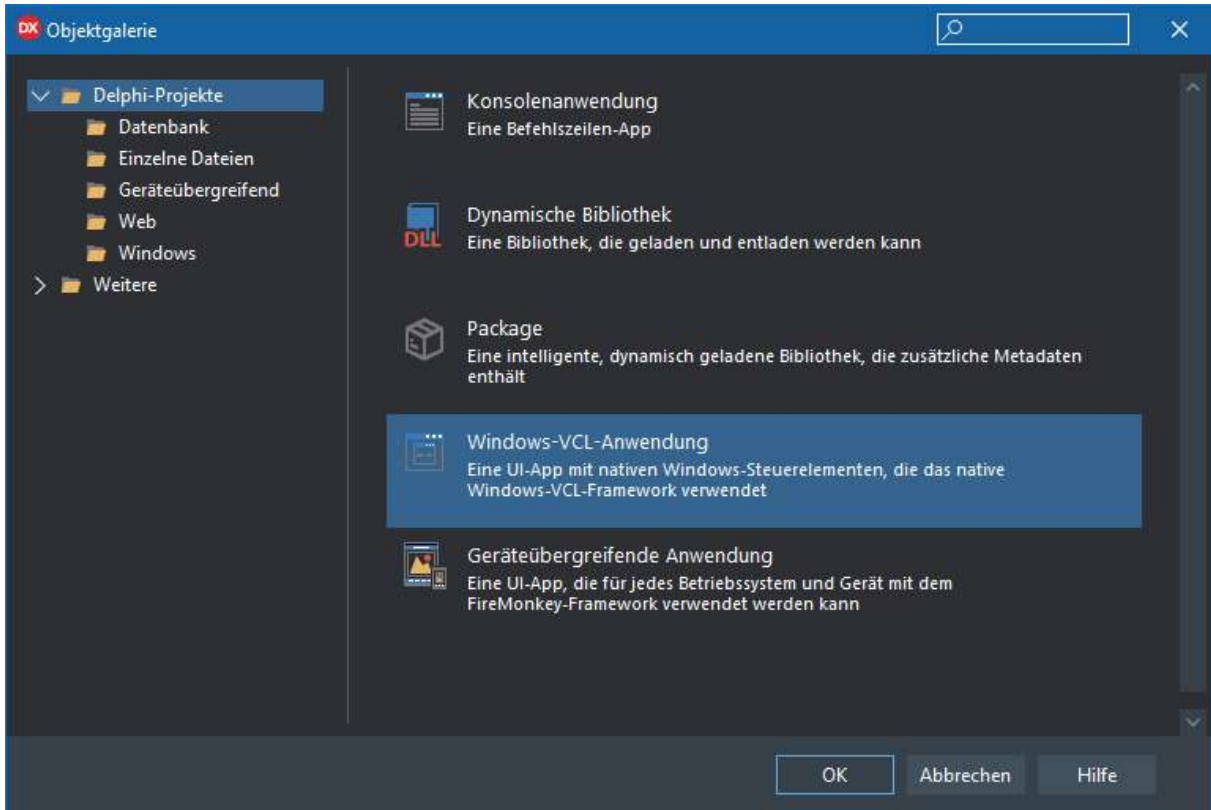
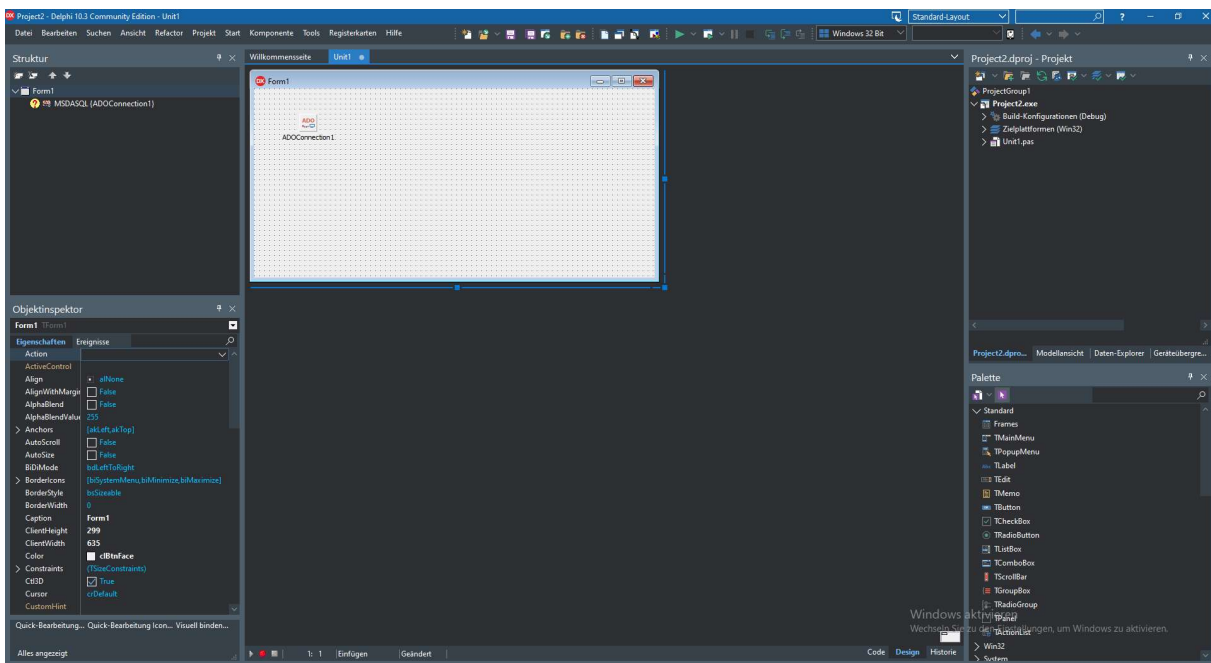


Delphi mit SAP ASE verbinden

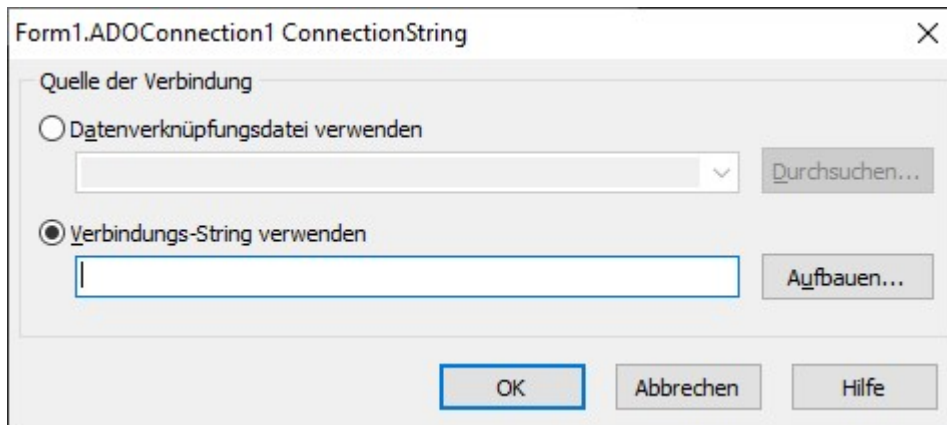
Diese Anleitung sollte es Ihnen ermöglichen über Delphi eine Verbindung zu einem SAP ASE Server aufzubauen. Sie öffnen dazu die Delphi-Umgebung und erstellen ein neues Projekt. Dies sollte eine VCL-Anwendung sein.



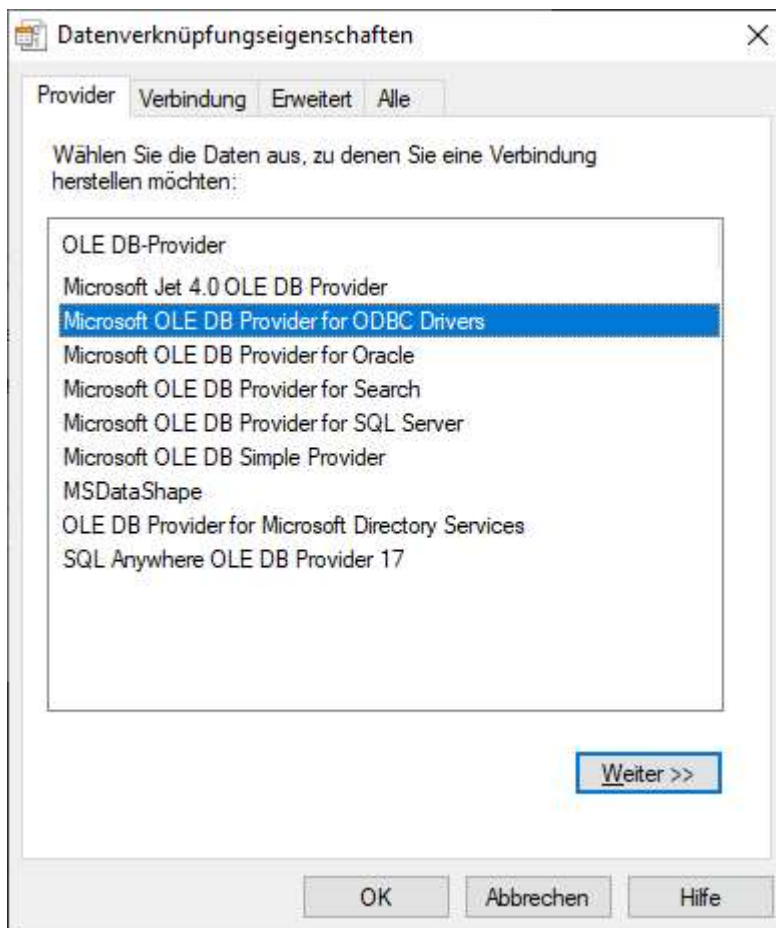
Sie sollten danach ein kleines Fenster mit dem Namen "Form1" sehen. In diesem können Sie eine "TADOConnection" hinzufügen. Diese finden Sie unten rechts in der Palette.



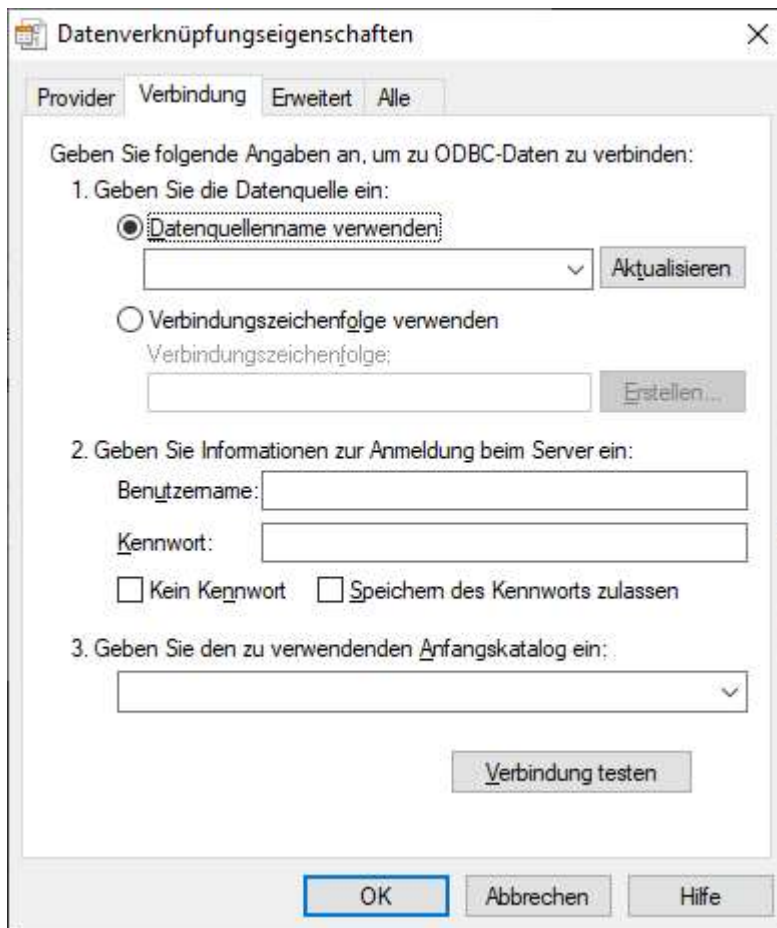
Nachdem Sie diese in das kleine Fenster hineingezogen haben, können Sie mit einem Doppelklick oder einem Rechtsklick auf die Komponenten und wählen "Verbindungs-String bearbeiten".



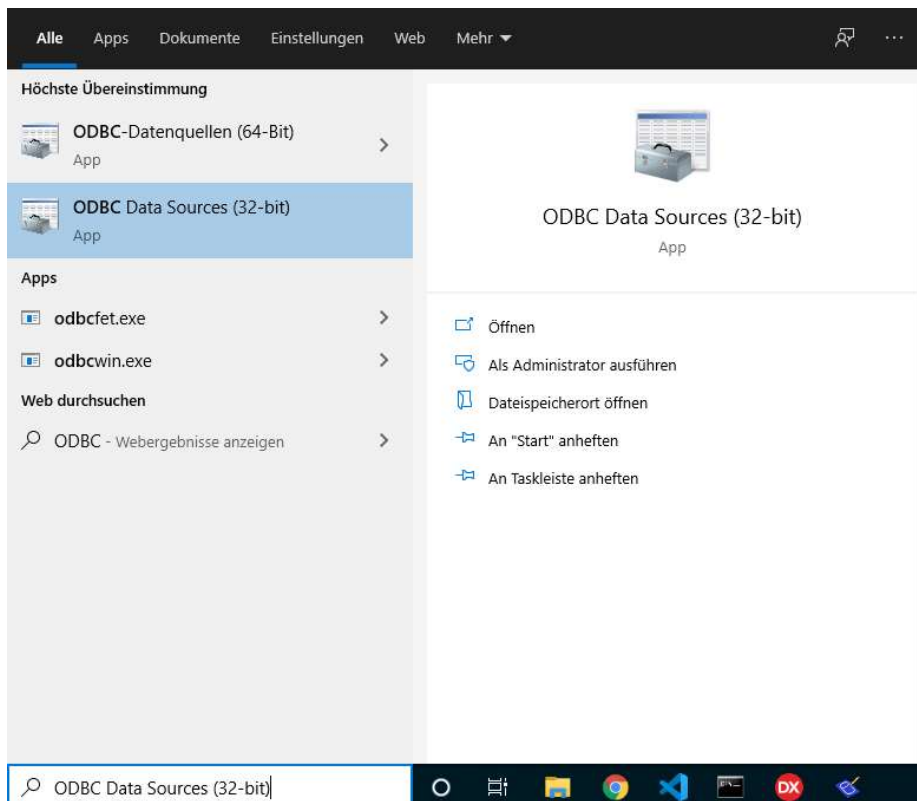
Dann klicken Sie auf den Knopf "Aufbauen...". Es öffnet sich ein neues Fenster, in dem Sie sich aussuchen können mit welcher Art Sie sich verbinden wollen. Um mit dem SAP ASE eine Verbindung aufzubauen wird der "Microsoft OLE DB Provider for ODBC Drivers" verwendet.



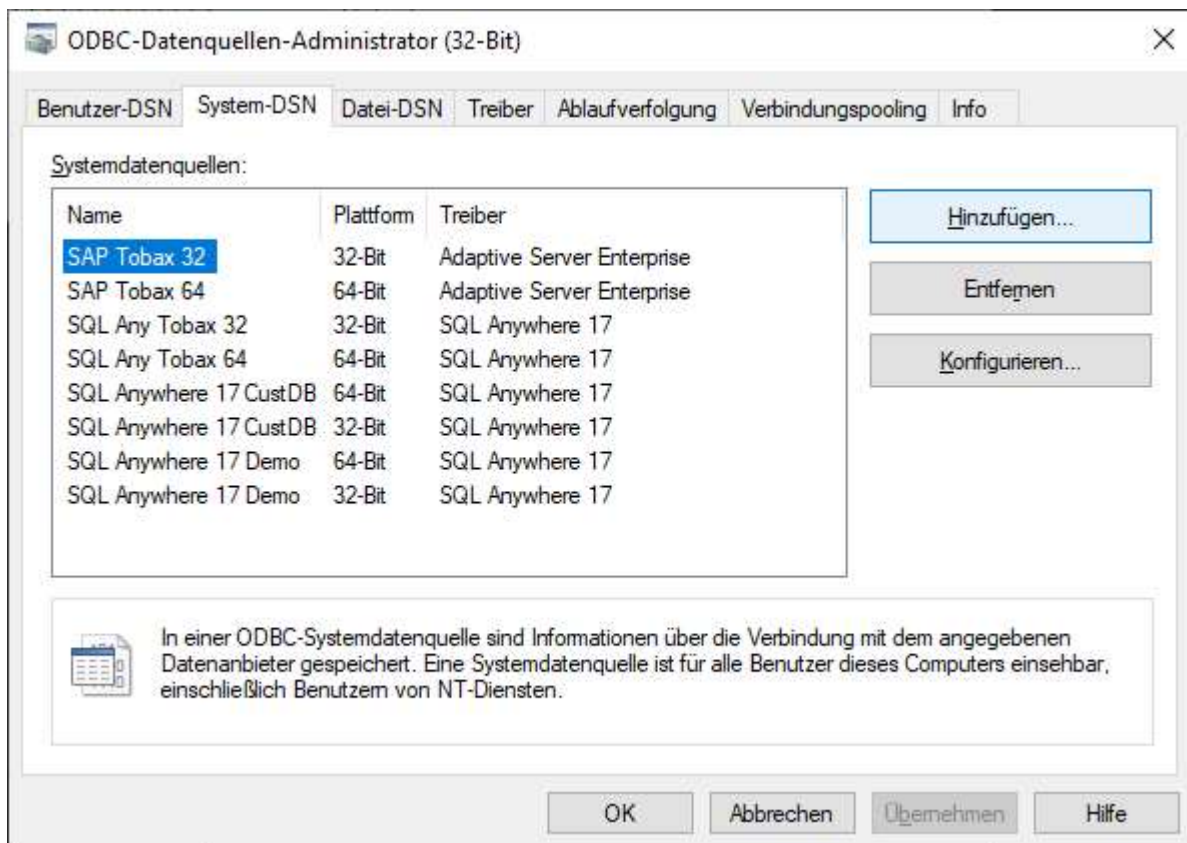
Drücken Sie anschließend auf den Knopf "Weiter>>". Der Reiter geht weiter zur Verbindung und Sie können sich eine ODBC-Datenquelle aussuchen, die der Connector verwenden soll.



Bevor wir hier weiter machen, müssen wir eine 32-Bit ODBC-Datenquelle erstellen. Suchen Sie mit Windows Start auf der Suchleiste "ODBC" und öffnen Sie den "ODBC Data Source (32-Bit)".



Wechseln Sie den Reiter von "Benutzer-DNS" auf "System-DNS" und fügen dort eine weitere DNS hinzu.



Wählen Sie bei der Datenquelle "Adaptive Server Enterprise" aus und klicken auf den Button "Fertig stellen".

The image shows a screenshot of the 'Adaptive Server Enterprise' configuration dialog box, specifically the 'Connection' tab. The dialog has a title bar with a question mark and a close button. Below the title bar are several tabs: 'General', 'Connection', 'Security', 'Advanced', 'Transactions', and 'About'. The 'Connection' tab is active. The main area contains several input fields and a dropdown menu:

- Data Source Name:** A text input field with a cursor.
- Description:** A text input field.
- Server Name (ASE Host Name):** A text input field.
- Server Port:** A text input field.
- Database Name:** A text input field.
- Logon ID:** A text input field.
- Service Name:** A text input field.
- BackEnd Type:** A dropdown menu with 'ASE' selected.

Below these fields is a section titled 'Cursor Behavior' containing a checkbox labeled 'Use Cursors'. To the right of this section is a 'Test Connection' button. At the bottom of the dialog are three buttons: 'OK', 'Abbrechen', and 'Übernehmen'.

Tragen Sie die Daten ein und benennen Sie die Datasource passend. Nachdem Sie alles eingestellt haben, können Sie eine Testverbindung erstellen.

Adaptive Server Enterprise

General Connection Security Advanced Transactions About

Data Source Name: SAP Tobax 32

Description:

Server Name (ASE Host Name): localhost

Server Port: 2638

Database Name: master

Logon ID: sa

Service Name: SAPTobax

BackEnd Type: ASE

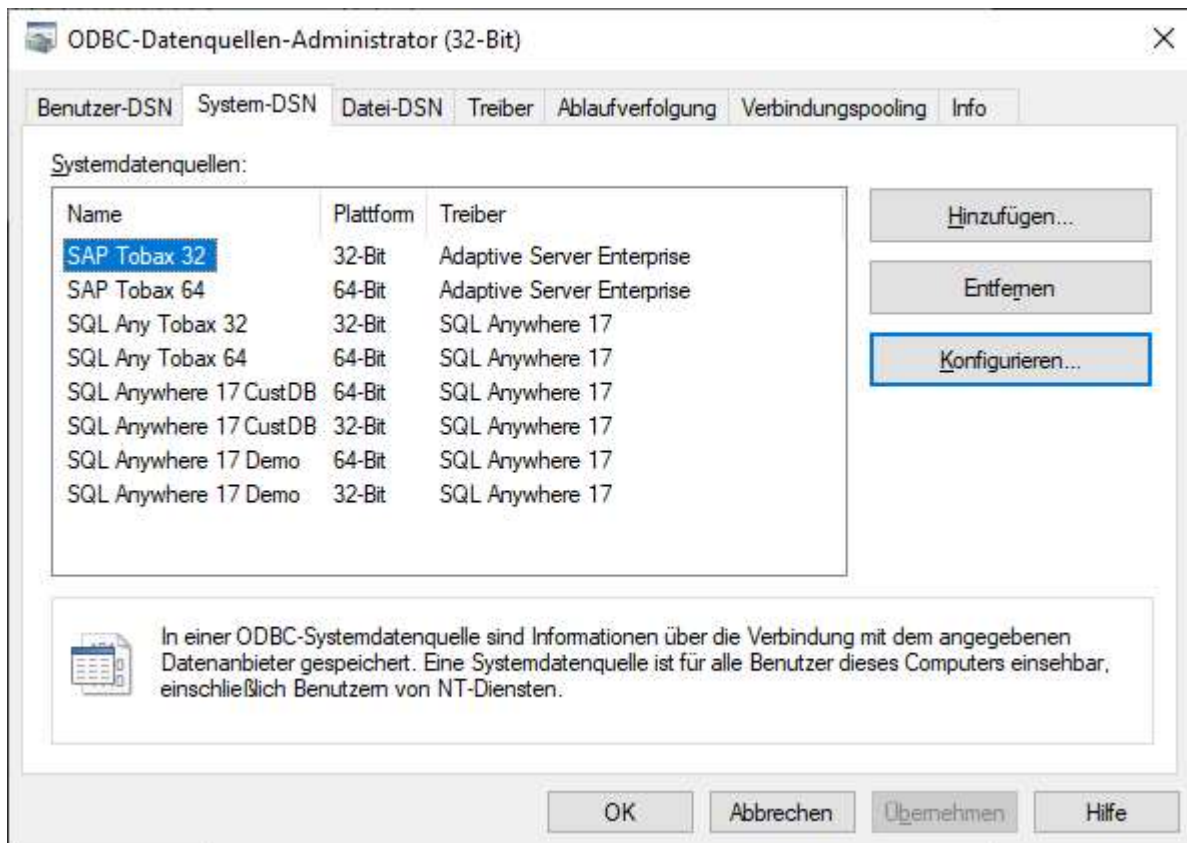
Cursor Behavior

Use Cursors

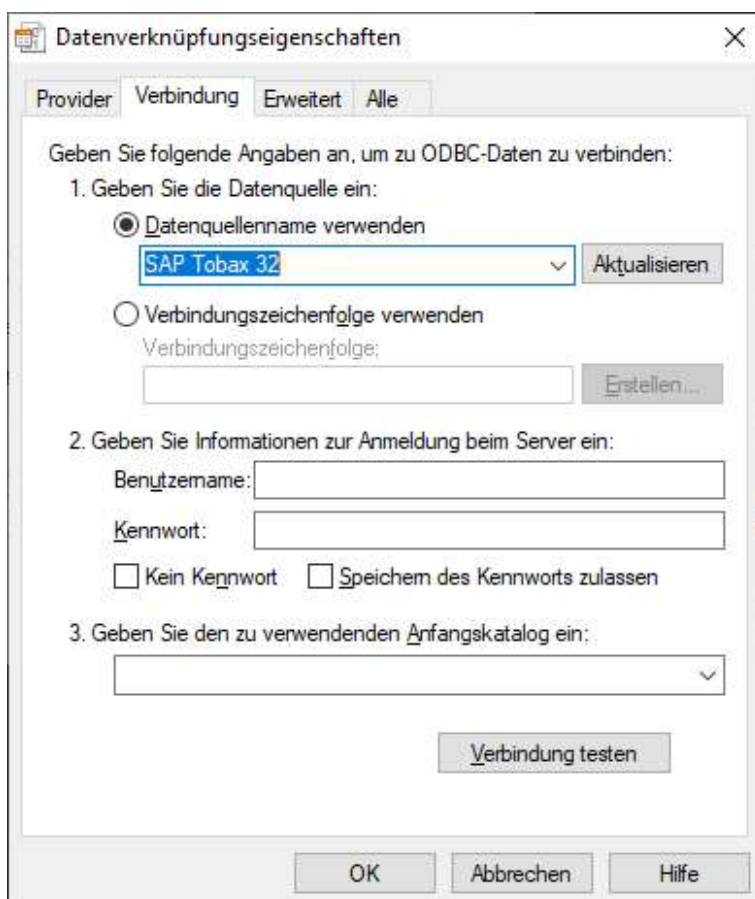
Test Connection

OK Abbrechen Übernehmen

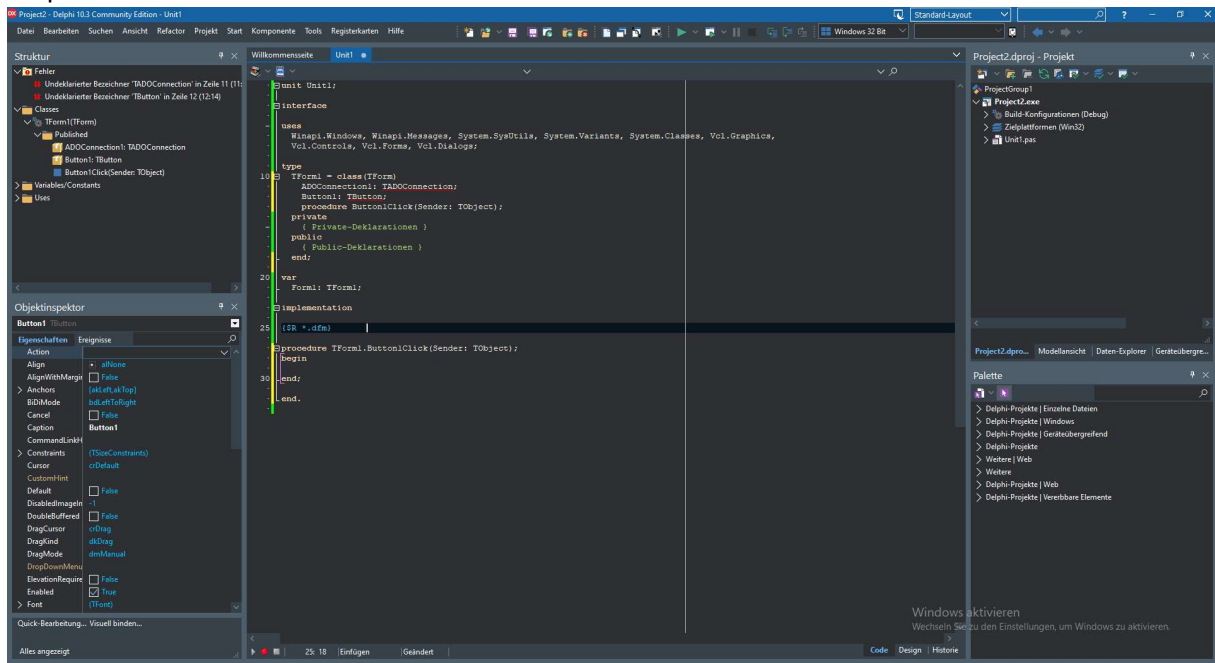
Damit die Datenquelle erstellt wird, drücken Sie auf den Knopf "OK". Danach sollte die ODBC-Datenquelle in der Liste erscheinen.



Sie können nun das Fenster mit den ODBC-Datenquellen schließen. Danach wechseln Sie zurück zu Delphi und aktualisieren die ODBC-Liste des Connectors. Die ODBC-Datenquelle sollte nun bei der Dropdown-Liste vorhanden sein. Wählen Sie diese anschließend aus.



Sie haben hier die Möglichkeit, bevor Sie das Fenster des Connector schließen, die ID und das Passwort für den SAP ASE einzutragen und zu testen. Nachdem Sie die Verbindung erstellt haben kopieren Sie diese in die Zwischenablage. Danach fügen Sie einen Knopf in das Form-Fenster von Delphi und wechseln die Ansicht zum Code.



Um über Delphi mit einem SAP ASE eine Verbindung aufzubauen, benutzen Sie folgenden Code:

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,  
Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Data.DB, Data.Win.ADODB;
```

```
type
```

```
TForm1 = class(TForm)  
ADOConnection1: TADOConnection;  
Button1: TButton;  
procedure Button1Click(Sender: TObject);
```

```
private
```

```
{ Private-Deklarationen }
```

```
public
```

```
{ Public-Deklarationen }
```

```
end;
```

```
const
```

```
ConnectionStr='Provider=MSDASQL.1;Persist Security Info=False;Data  
Source=<Datenquellennamen>;Port=<Portnummer>;Uid=<BenutzerID>;Pwd=<Passwort>';
```

```
var
```

```
Form1: TForm1;
```



```
AdoConnection: TADOConnection;
```

implementation

```
{$R *.dfm}
```

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

```
procedure ConnectServer(DataBase:string);
```

```
begin
```

```
  ShowMessage('Verbindung zum Server wird hergestellt');
```

```
  AdoConnection:=TADOConnection.Create(nil);
```

```
  AdoConnection.LogInPrompt:=False;
```

```
  AdoConnection.ConnectionString:=Format(ConnectionStr,['Tobax',dataBase]);
```

```
  AdoConnection.Connected:=True;
```

```
end;
```

```
procedure ReadData;
```

```
var AdoQuery: TADOQuery;
```

```
begin
```

```
  AdoQuery:=TADOQuery.Create(nil);
```

```
  try
```

```
    AdoQuery.Connection:=AdoConnection;
```

```
    AdoQuery.SQL.Add('SELECT * FROM <Tabelle>');
```

```
    AdoQuery.Open;
```

```
    while not AdoQuery.eof do
```

```
      begin
```

```
        ShowMessage(Format('%<Datentyp> %<Datentyp>  
%<Datentyp>', [AdoQuery.FieldByName('<Spalte>').asInteger, AdoQuery.FieldByName('<Spalte>').asStr  
ing, AdoQuery.FieldByName('<Spalte>').asString]));
```

```
        AdoQuery.Next;
```

```
      end;
```

```
    finally
```

```
      AdoQuery.Free;
```

```
    end;
```

```
end;
```

```
procedure disconnectServer;
```

```
begin
```

```
if ADOConnection.Connected then
```

```
  AdoConnection.Close;
```

```
  AdoConnection.Free;
```

```
  ShowMessage('Verbindung zum Server getrennt');
```

```
end;
```

```
begin
```

```
  try
```

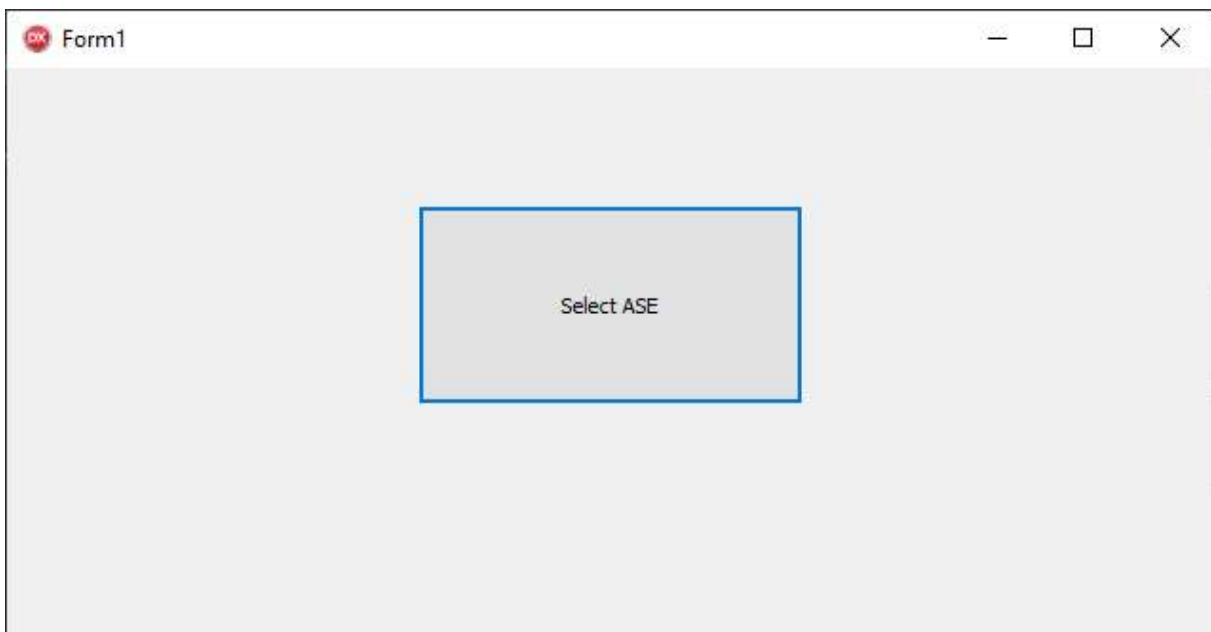
```
    ShowMessage('Init');
```

```
    try
```

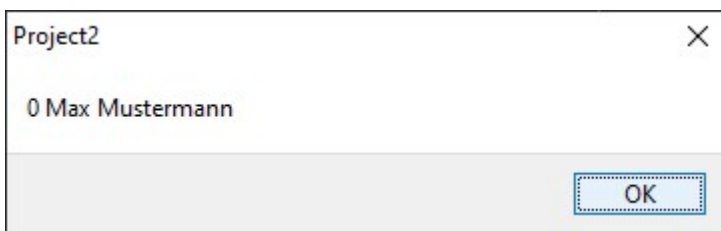
```
      ConnectServer('<Database>');
```

```
ReadData;  
disconnectServer;  
except  
on E : Exception do  
ShowMessage(E.Classname + ' ' + E.Message);  
end;  
finally  
end;  
end;  
end.  
  
end.
```

Nachdem Sie den Code für ihre Datenbank ergänzt haben, können Sie diesen testen. Nutzen Sie oben auf der Leiste die grüne Abspieltaste, um das Projekt zu erzeugen.



Drücken Sie auf den Knopf und kontrollieren Sie, ob die Daten angezeigt werden.



Werden Ihnen die Message Dialoge angezeigt und können Sie die Testdaten, die Sie erzeugt haben sehen, hat alles wie gewünscht funktioniert.

Bei Fragen:
y.duong@tobax.de
Yao Zu Duong

Tel. 02171 3606 19

www.tobax.de