

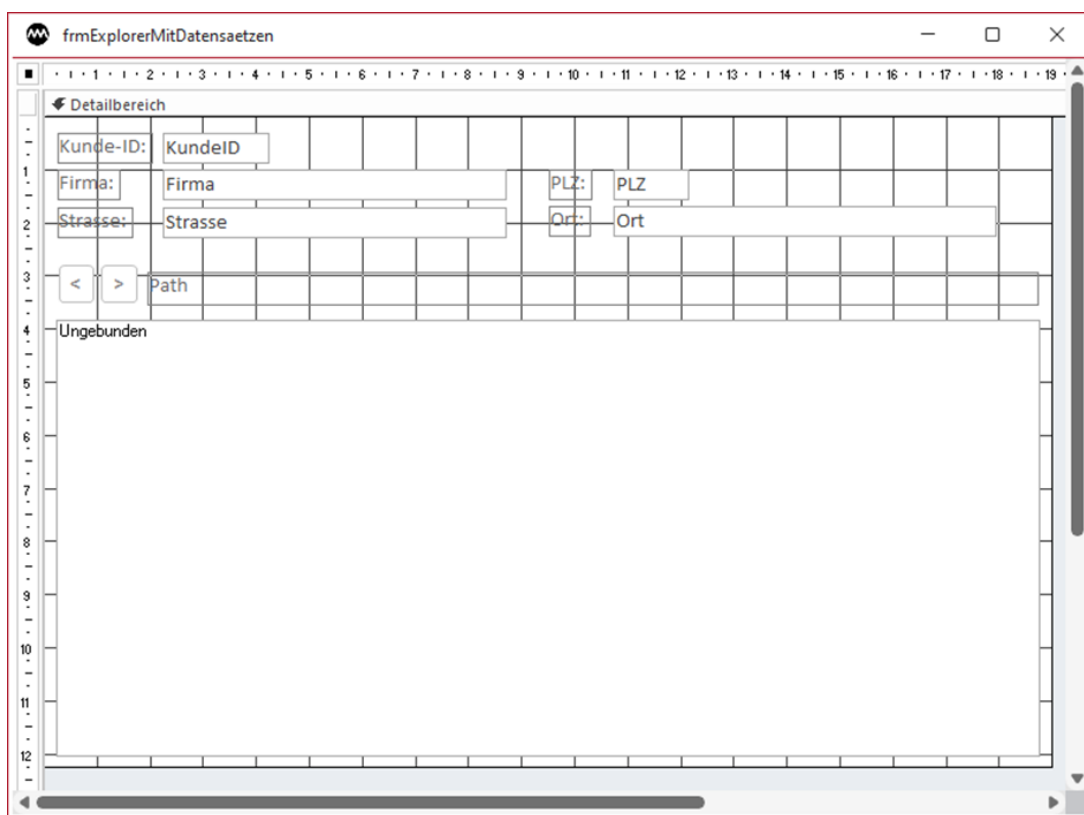


amvExplorer mit verschiedenen Ordnern je Datensatz

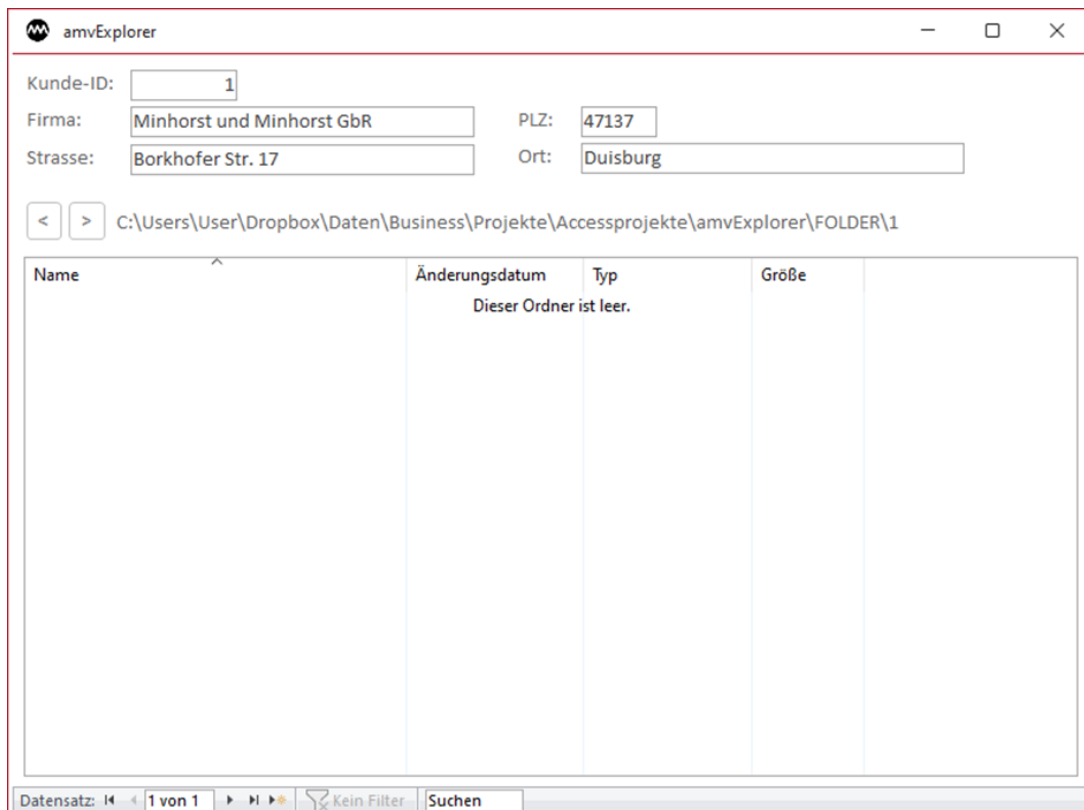
André Minhorst - 2024-03-10 - Comments (0) - amvExplorer

amvExplorer mit verschiedenen Ordnern je Datensatz

Wir wollen nun, dass für jeden Datensatz ein Ordner angelegt und angezeigt wird. Dazu binden wir ein Formular an die Tabelle **tblKunden** und fügen die Felder dieser Tabelle hinzu. Außerdem legen wir wieder die Elemente aus dem vorherigen Beispiel an.



Damit können wir nun zu jedem Kunden einen eigenen Ordner anzeigen. Diese wird nach der Kunden-ID benannt und jeweils mit dem Kunden angezeigt:



Dazu ist ein wenig mehr Code nötig.

Wir verwenden als Basis den Code des vorherigen Beispiels:

```
Dim WithEvents objExplorer As CLASS_AMV_EXPLORER
```

```
Private Sub Form_Open(Cancel As Integer)
    Set objExplorer = New CLASS_AMV_EXPLORER
    With objExplorer
        'Version 1.0.0.0
        .Initialize Me.ctlExplorer, CurrentProject.Path, Me.cmdBack,
        _
        Me.cmdForward, Me.lblPath
        'Version 1.0.0.1
        .Initialize Me.Form, Me.ctlExplorer, CurrentProject.Path,
        Me.cmdBack, _
        Me.cmdForward, Me.lblPath
    End With
End Sub
```

```
Private Sub cmdOK_Click()
    DoCmd.Close acForm, Me.Name
End Sub
```

Dann müssen wir berücksichtigen, ob gerade ein neuer Datensatz oder ein bestehender Datensatz angezeigt wird. Das machen wir in Beim Anzeigen:

```
Private Sub Form_Current()  
    If Me.NewRecord Then  
        Call TempPath  
    Else  
        Call SetPath  
    End If  
End Sub
```

Bei einem neuen Datensatz soll ein Verzeichnis namens **Temp** angezeigt werden. Wir prüfen, ob dieses vorhanden ist, legen es gegebenenfalls an und zeigen es dann an:

```
Private Sub TempPath()  
    If Not CBool(PathFileExists(CurrentProject.Path &  
"\FOLDER\Temp")) Then  
        Mkdir CurrentProject.Path & "\FOLDER\Temp"  
    End If  
  
    If objExplorer.Path <> CurrentProject.Path & "\FOLDER\Temp" Then  
        objExplorer.Path = CurrentProject.Path & "\FOLDER\Temp"  
    End If  
End Sub
```

Wenn hingegen ein vorhandener Datensatz angezeigt wird, prüfen wir, ob schon ein Verzeichnis mit dem Namen des Feldwertes aus **KundeID** vorhanden ist und legen es gegebenenfalls an, danach blenden wir es ein:

```
Private Sub SetPath()  
    If Not CBool(PathFileExists(CurrentProject.Path & "\FOLDER\" &  
Me.KundeID)) Then  
        Mkdir CurrentProject.Path & "\FOLDER\" & Me.KundeID  
    End If  
    If objExplorer.Path <> CurrentProject.Path & "\FOLDER\" &  
Me.KundeID Then  
        objExplorer.Path = CurrentProject.Path & "\FOLDER\" &  
Me.KundeID  
    End If  
End Sub
```

Schließlich fehlt noch eine Aktion für das Anlegen eines neuen Datensatzes. Sobald dieser gespeichert wird, feuert das Ereignis **Nach Aktualisierung**, das wir wie folgt nutzen, um den Ordner anzulegen und anzuzeigen:

```
Private Sub Form_AfterUpdate()  
    Call SetPath  
End Sub
```

Wichtiger Hinweis

Man kann nicht von der Formularansicht zum Entwurf wechseln, wenn die hier vorgestellte Technik implementiert ist. Man muss dann das Formular schließen und dieses dann im Entwurf öffnen.

Hinweis

Achtung: Die für diese Anleitung nötige Beispieldatenbank erhältst Du nach dem Kauf des **amvExplorers**:

[Zum Shop ...](#)

Hast Du den **amvExplorer** bereits gekauft, findest Du diesen nach der Anmeldung im Shop unter **Mein Konto|Sofortdownloads**.