

Create-LV-Pgm

Das Programm **Create-LV-Pgm** ermöglicht mit wenigen Eingaben die Erstellung eines funktionsfähigen Profan Programmes zur Anzeige einer .csv-Datei.

Es wird die **ListView.dll** von **Frank Abbing** benutzt.

Das erstellte Programm kann jederzeit erweitert werden um z.B. die angezeigte Datei auch bearbeiten zu können.

In der erstellten Form erfolgt nur die Anzeige der Datei, inklusive blättern und Auswahl einer Zeile mittels Doppelklick.

Das Programm kann aber abgeändert werden, sodass die Auswahl auch mit einem Einfachklick erfolgt.

Bei den anzuzeigenden **.csv-Dateien** gibt es 2 Arten:

- In der Datei ist der erste Satz ein **Header-Satz**. Hier sind die Spaltenüberschriften angegeben. Ab dem zweiten Satz beginnen die Datensätze.
- Es gibt **keinen Header-Satz**. Die Daten beginnen sofort mit dem ersten Datensatz. Für diese Dateiart müssen die Spaltenüberschriften eingegeben werden.

Eine .csv-Datei ist im Prinzip eine normale Textdatei mit variabler Satzlänge. Zwischen den Spaltentexten befindet sich ein Trennzeichen. Dieses ist meistens ein Komma oder ein Semikolon. Es gibt aber auch Versionen die ein Tabulatorzeichen oder irgendein bestimmtes Zeichen verwenden.

Inhalt

[Installation](#)

[Ablauf einer Programm Erstellung](#)

[Nachbearbeitung des erstellten Programmes](#)

[Bemerkungen zur ListView.dll und zum Programm SumatraPDF](#)

Zur Hilfeanzeige wird im Programm **Create-LV-Pgm** anstelle des **Acrobat Readers** das schnellere Freeware-Programm **SumatraPDF** benutzt.

Unabhängig davon kann die Hilfedatei **Create-LV-Pgm.pdf** ausserhalb des Programmes **Create-LV-Pgm** auch mit dem **Acrobat Reader** geöffnet werden.

Die **ListView.dll** ist auf [Frank Abbing's Webseite](#) zu finden.

Der Freeware PDF Betrachter **SumatraPDF** kann u,a, von [hier](#) heruntergeladen werden.

Programmbeschreibung Create-Lv-Pgm

Installation

Erstellen Sie ein Verzeichnis und entpacken Sie die Zip-Datei in dieses Verzeichnis.

Da für dieses Programm und auch später für die erstellten Programme die Freeware **ListView** von Frank Abbing benötigt wird laden Sie diese von der Webseite von [Frank Abbing](#) herunter und installieren Sie diese.

Die Freeware **SumatraPDF** liegt diesem Paket bei. Diese Version ist im Moment die neueste Version und funktioniert mit dem Programm **Crt-Lv-Pgm**. Bei künftigen Versionen von **SumatraPDF** kann ich nicht abschätzen ob die Funktion mit **Crt-Lv-Pgm** auch gegeben ist.

Obwohl die **ListView.dll** ebenfalls beigelegt ist (damit **Crt-Lv-Pgm** ausgeführt werden kann) sollte das Freewareprogramm **ListView** doch installiert werden.

Ablauf einer Programm Erstellung

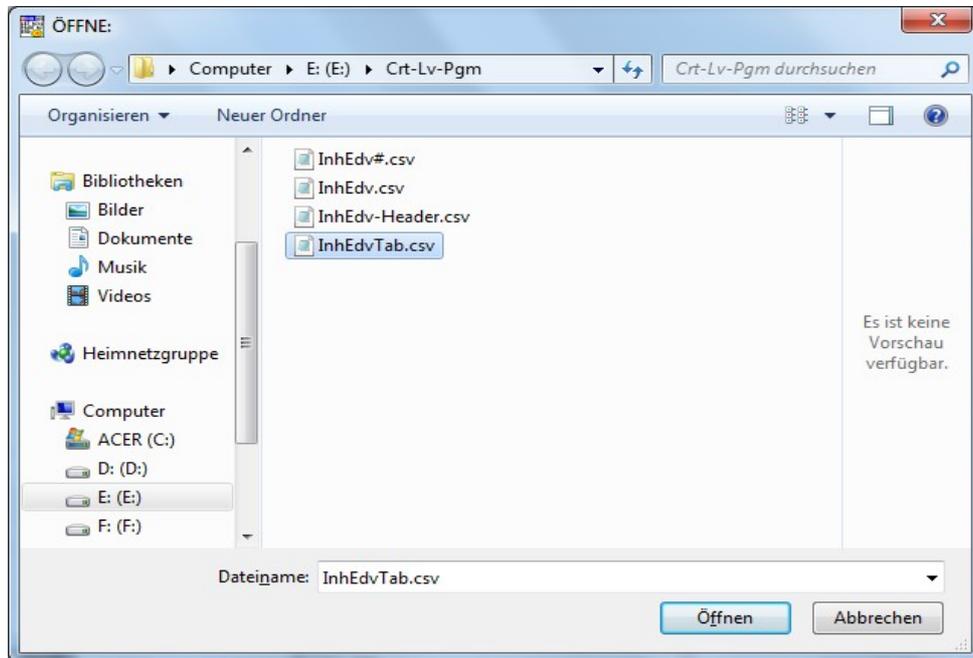


Bild 01.

Nach dem Start des Programmes **Crt-Lv-Pgm** wird sofort die Dateiauswahl angezeigt.

Nach der Auswahl der Datei **InhEdvTab.csv** erscheint der Dialog zur Auswahl des Zeichens der Spaltentrennung. Eine .csv Datei hat normalerweise ein Komma um die Texte der einzelnen Spalten zu trennen. Abgesehen vom Komma ist auch manches mal ein Semikolon oder ein Tabulatorzeichen. Es gibt aber noch andere Zeichen.

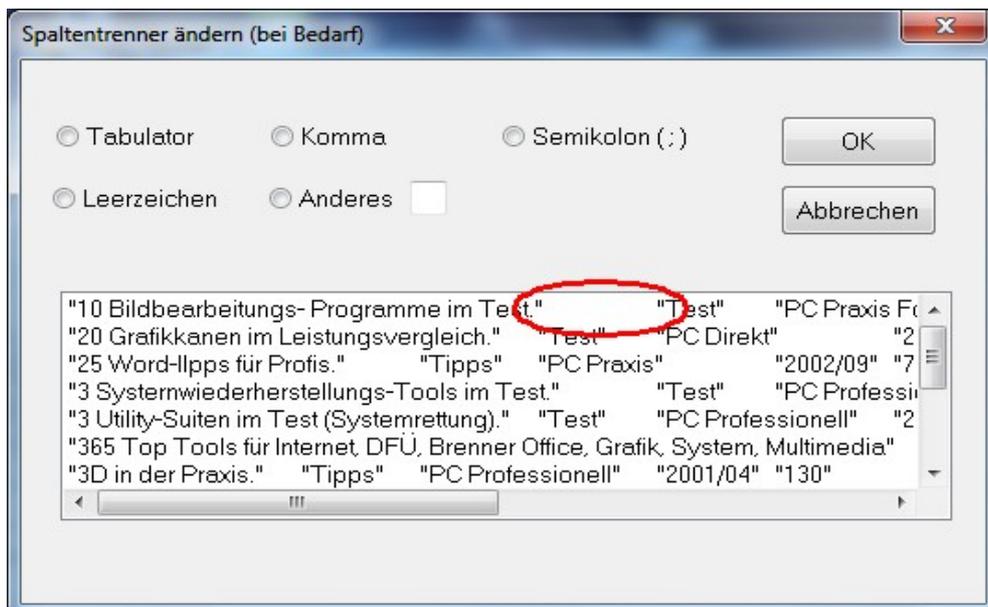


Bild 02.

Hier wurde offensichtlich eine Datei die einen Tabulator als Trennzeichen hat gewählt (erkennbar an den unterschiedlich langen Leerstellen im Text). Sonst wäre nur ein Komma, Semikolon ... anstelle der Leerstellen zu sehen.

Um die Entscheidung, um welches Zeichen es sich handelt zu treffen, können die angebotenen Vorgaben (Tabulator, Komma etc.) ausprobiert werden.

Programmbeschreibung Create-Lv-Pgm



Bild 03.

Wenn das Bild, ähnlich wie abgebildet, erscheint, wo die Texte der einzelnen Spalten plausibel aussehen, wurde der richtige Separator (hier mit dem Zeichen `"` dargestellt) gefunden. Mit OK wird dieser Dialog beendet.

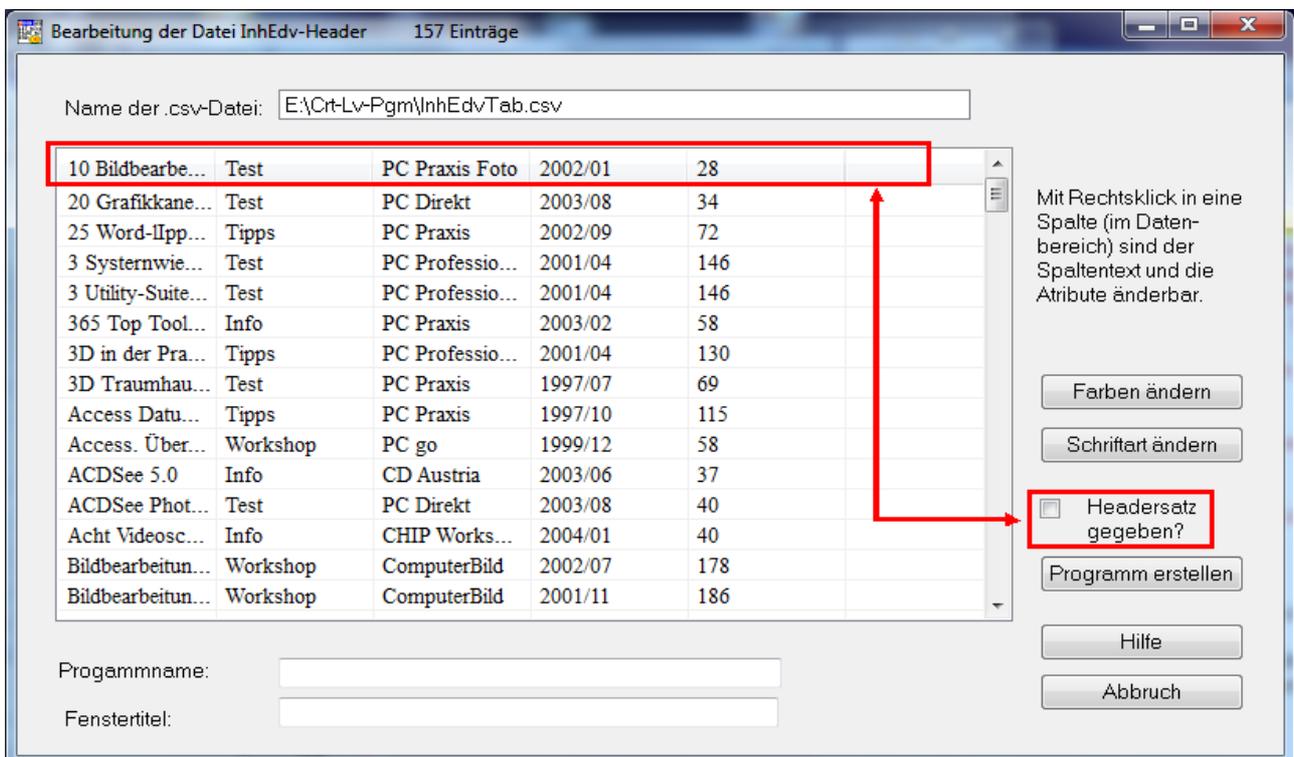


Bild 04.

Und in dieser Form sollte die Datei angezeigt werden. Die Spalten sind noch auf eine fixe Breite von 100 Punkten vorgegeben. Das kann später geändert werden.

Wenn man hier die Texte in den Spaltenbuttons ansieht ist zu erkennen, dass hier kein Headersatz in der Datei existiert (die Texte in den Spaltenbuttons lassen erkennen, dass es sich bereits um den ersten Datensatz handelt). Daher ist es bei dieser Datei wichtig Headersatz gegeben? Nicht zu aktivieren.

Programmbeschreibung Create-Lv-Pgm

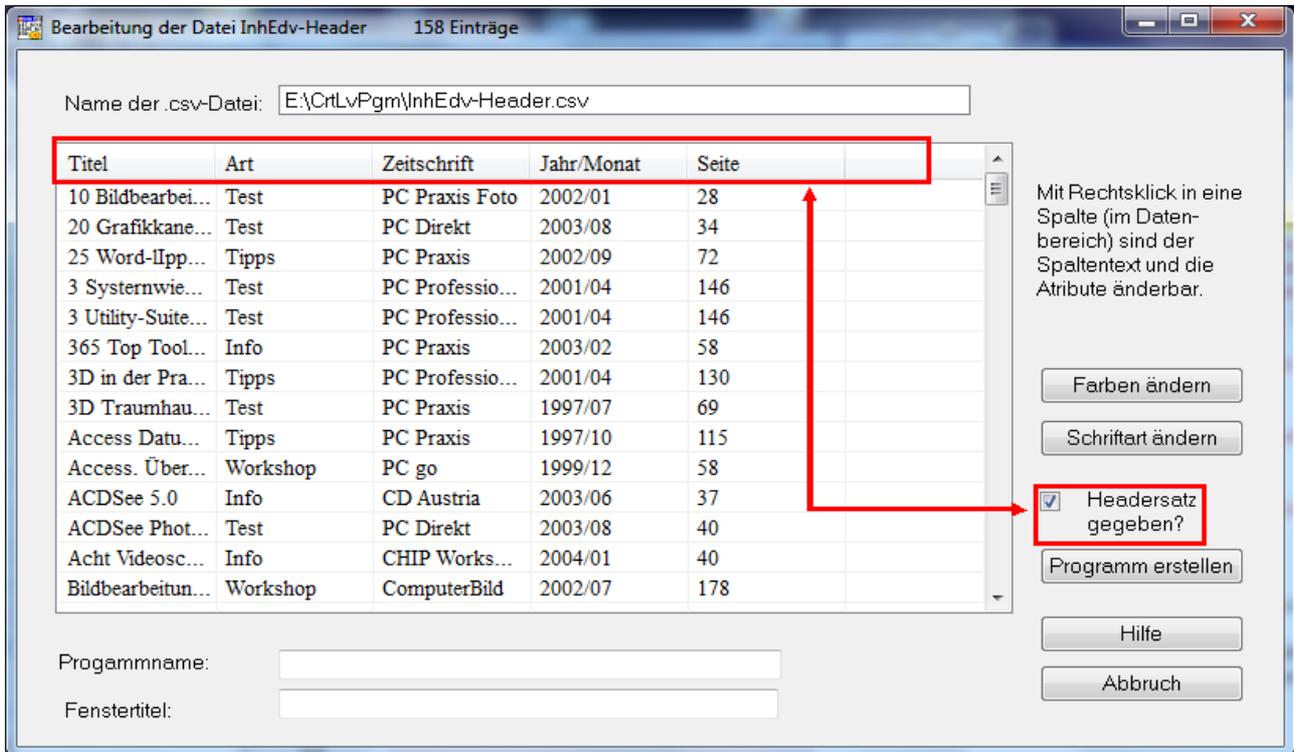


Bild 05.

Hier wurde eine andere Datei geladen.

Wichtig ist hier, dass die Spaltenbuttons offensichtlich anzeigen, dass die Datei einen Headersatz enthält. Daher ist es wichtig [Headersatz gegeben?](#) zu aktivieren.

Die weitere Verarbeitung ist für alle Dateiartern (mit/ohne Headersatz, Spaltentrennzeichen ...) gleich.

Zuerst sollten die Spaltenattribute korrigiert werden. Mit Rechtsklick in eine Spalte (nicht auf den Spaltenbutton selbst) wird ein Dialog zum [Spalten bearbeiten](#) geöffnet.

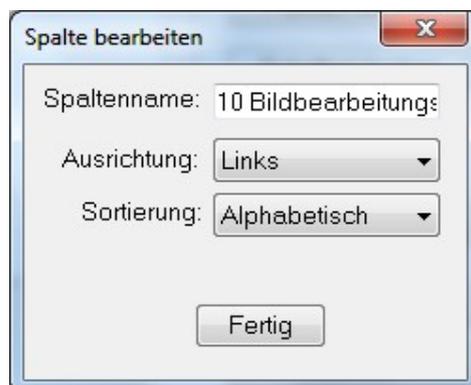


Bild 06.

Hier kann der Spaltenname geändert werden (im Beispiel in Bild 04 erforderlich weil hier die Spaltennamen aus dem ersten Datensatz interpretiert werden). Keine Angst, der erste Datensatz wird nicht in der Datei verändert.

Weiters kann die Textausrichtung in der Spalte geändert werden. Bei numerischen Feldern (wie hier im Beispiel in der 5. Spalte) ist [rechts](#)(bündig) zweckmäßig. Auch die Sortierung (wenn der Spaltenknopf gedrückt wird) sollte auf [numerisch](#) eingestellt werden.

Programmbeschreibung Create-Lv-Pgm



Bild 07.

Für diese 5. Spalte habe ich diese Einstellungen gewählt.

Nachdem in den Spalten 1 bis 4 nur der Spaltenname zu ändern war (Ausrichtung und Sortierung blieben unverändert) erscheint die Anzeige so:

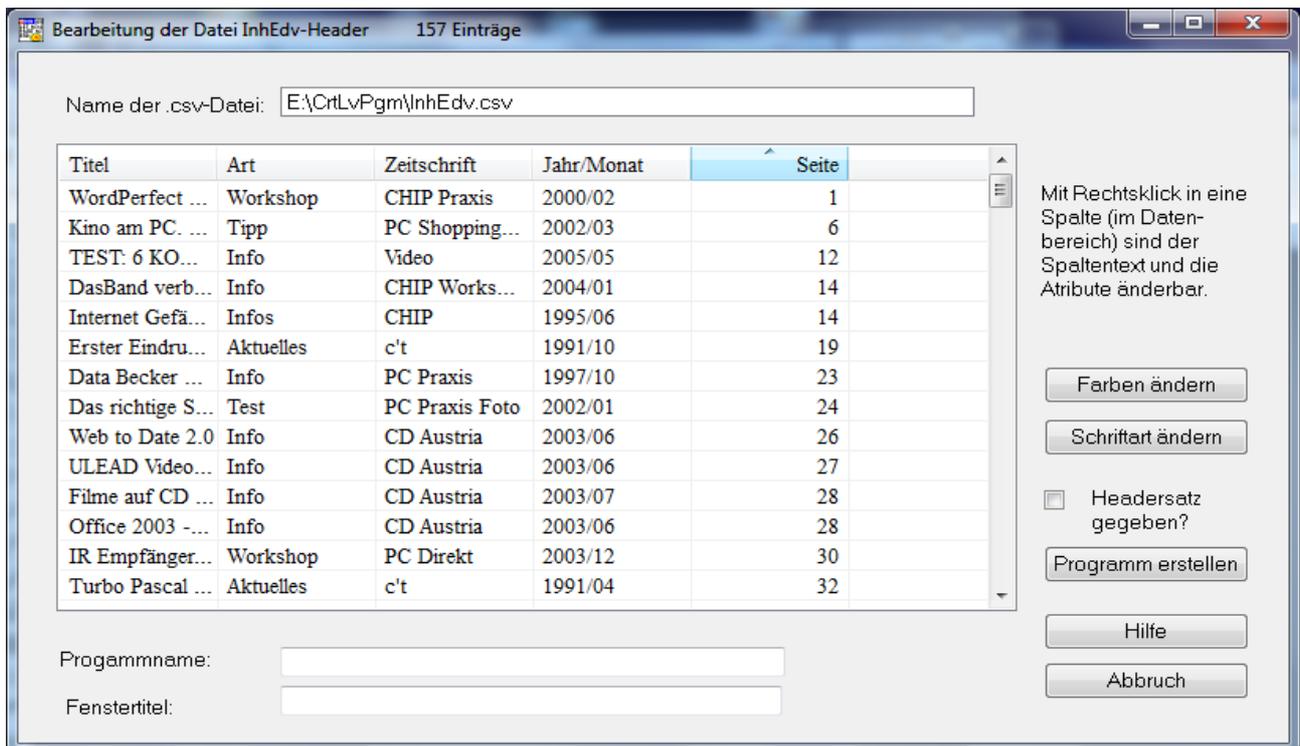


Bild 08.

Wenn die Ausrichtung der 5. Spalte nicht ganz korrekt erscheint wird sie nach einem Klick auf den **Spaltenbutton Seite** aufsteigend sortiert und gleichzeitig ausgerichtet.

Egal wie hier sortiert wird hat es keinen Einfluss darauf wie dann im erstellten Programm die Daten angezeigt werden. Die Anzeige ist am Beginn immer in der Reihenfolge der Sätze in der Datei!

Programmbeschreibung Create-Lv-Pgm

Jetzt soll die Schriftart geändert werden. Dies ist nur wahlweise – erforderlich ist das nicht.

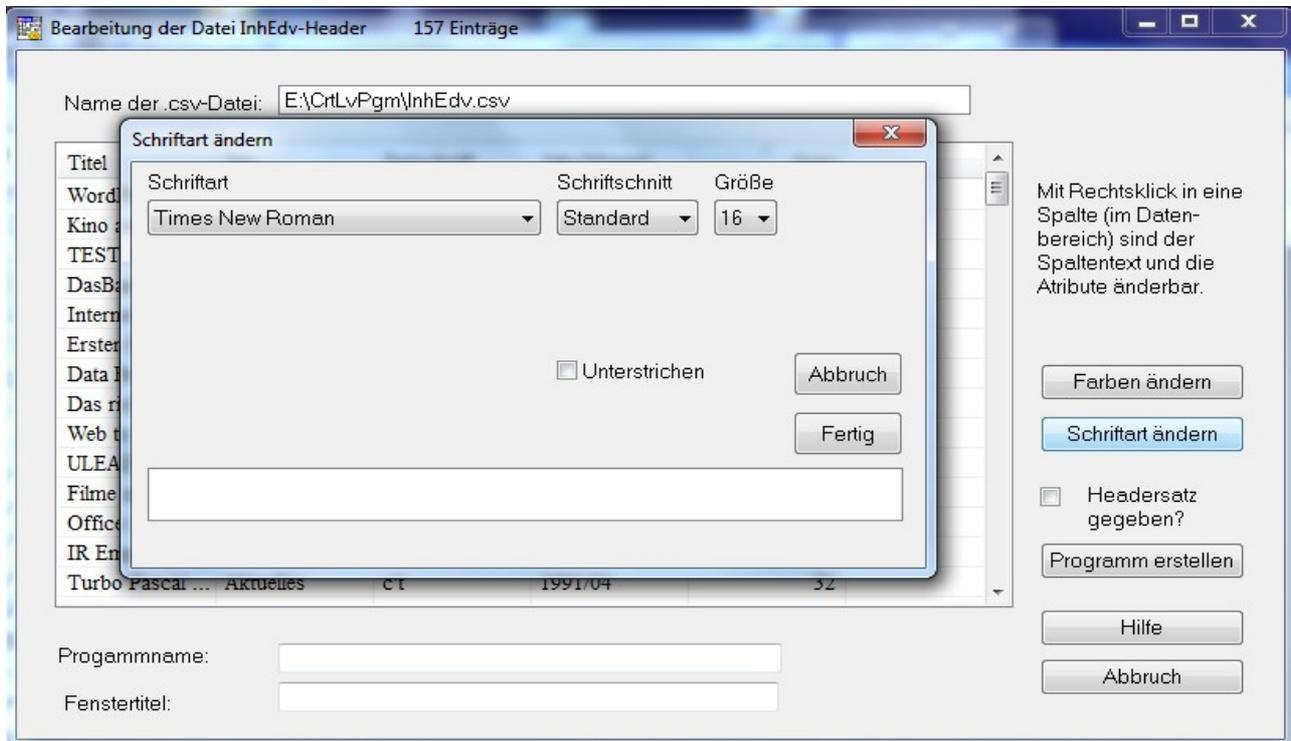


Bild 09.

Mit Klick auf den Button **Schriftart ändern** wird dieser Dialog angezeigt.

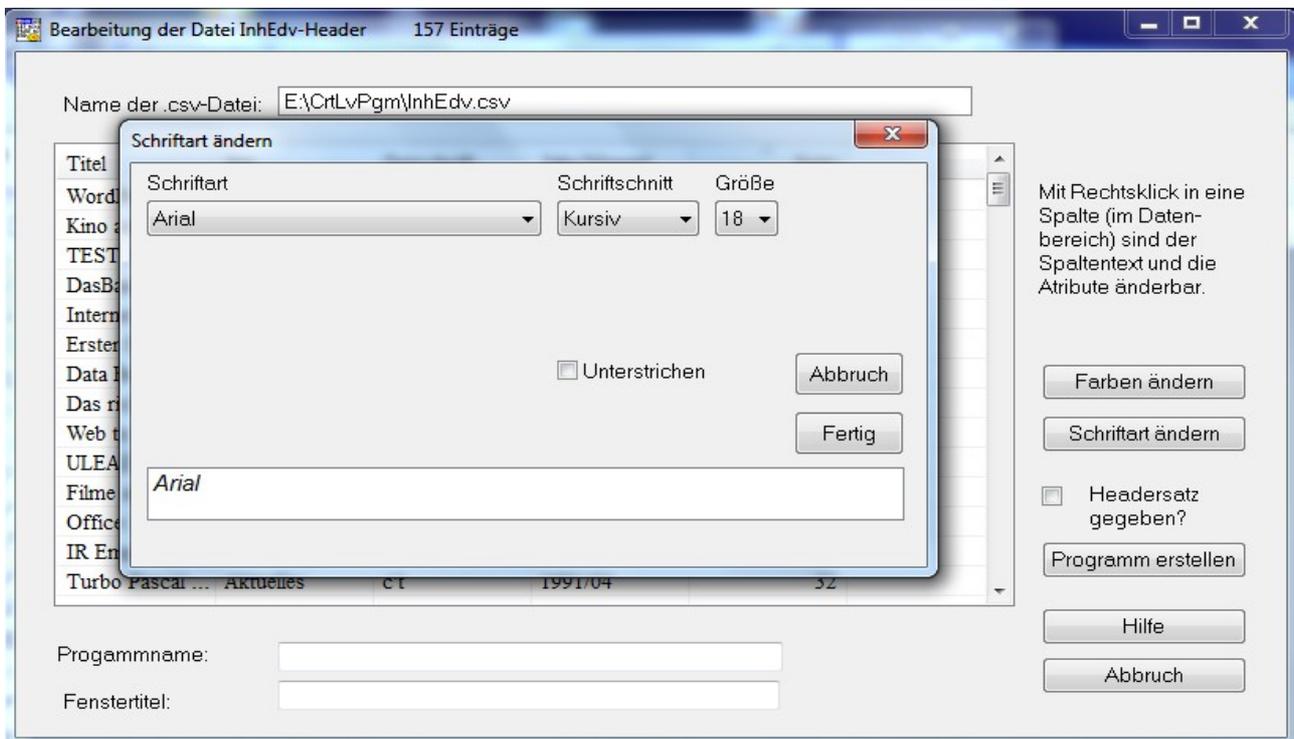


Bild 10.

Wird eine andere Schriftart und/oder Schriftschnitt und/oder Größe geändert wird das Schriftbeispiel sofort in der Zeile darunter angezeigt.

Auch unterstreichen ist möglich. Das ist aber optisch nicht sehr sinnvoll.

Programmbeschreibung Create-Lv-Pgm

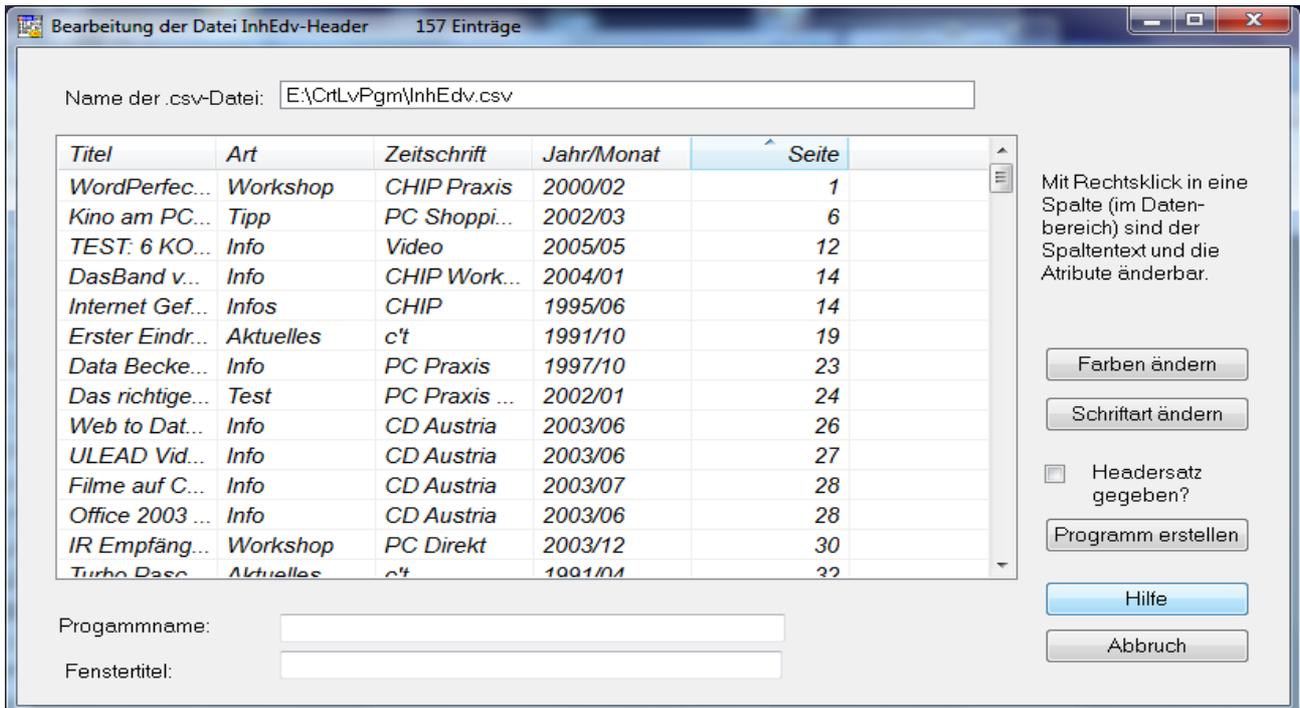


Bild 11.

Wenn die Ansicht nicht ganz entspricht kann die Schriftart wieder geändert werden. Eine andere Schriftfarbe ist bei diesem Dialog nicht vorgesehen. Dies ist bei **Farben ändern** möglich.

Mit Klick auf den Button **Farben ändern** wird der entsprechende Dialog angezeigt.

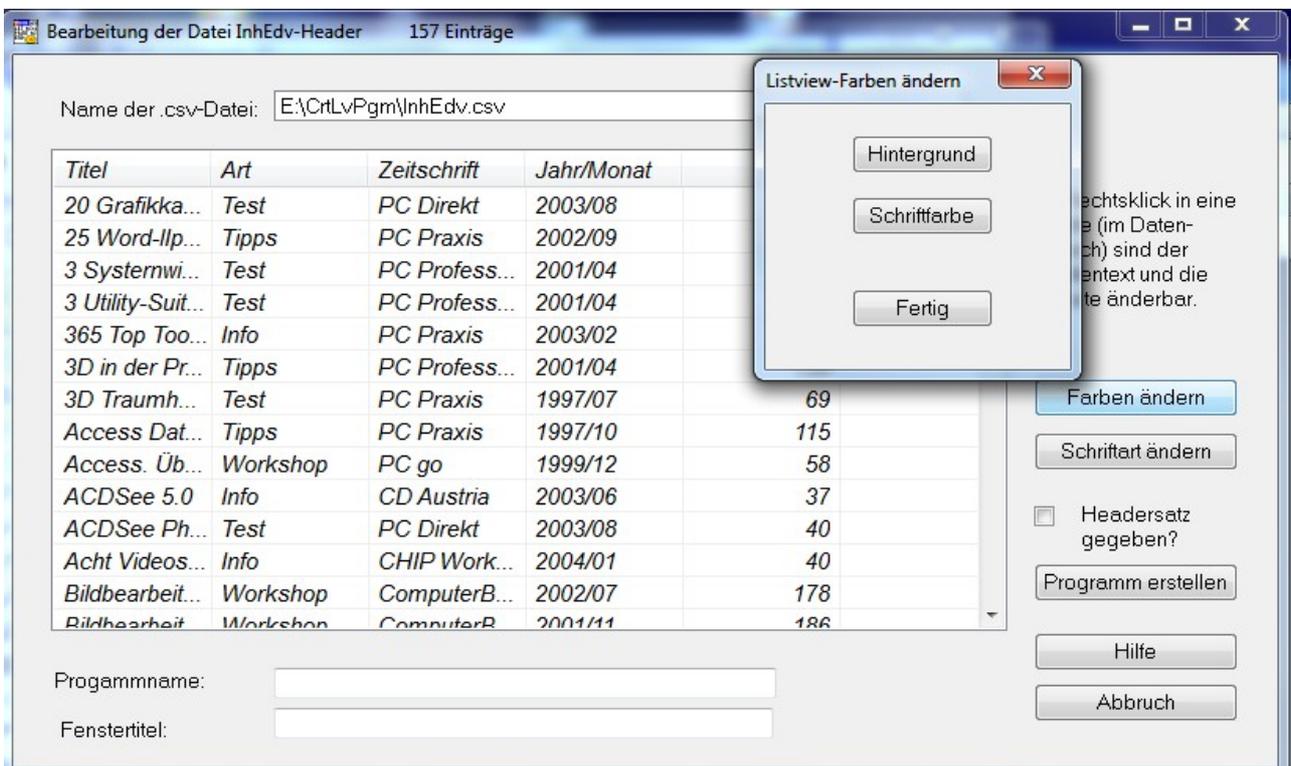


Bild 12.

Die Farbänderung kann sowohl nur den Hintergrund oder nur die Schriftfarbe als auch beide Möglichkeiten betreffen.

Programmbeschreibung Create-Lv-Pgm

Als Beispiel habe ich hier **Hintergrund** gewählt,

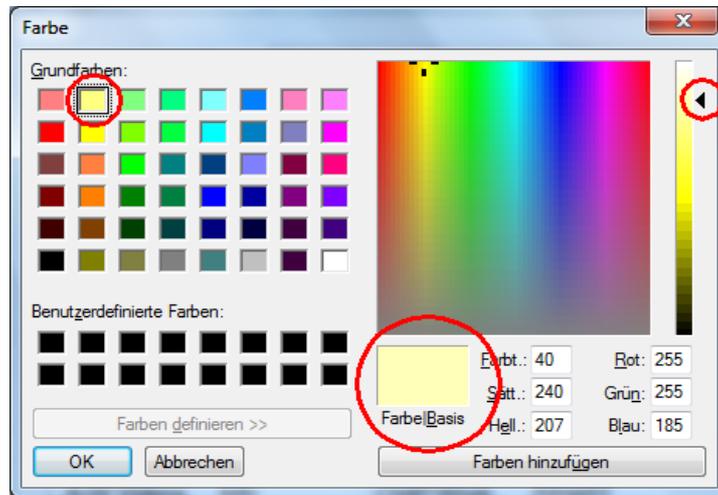


Bild 13.

Zuerst auf eine Grundfarbe klicken, dann am rechten senkrechten Balken den gewünschten Farbton einstellen, der dann im kleinen Rechteck angezeigt wird. Mit **OK** und **Fertig** wird der Dialog beendet.

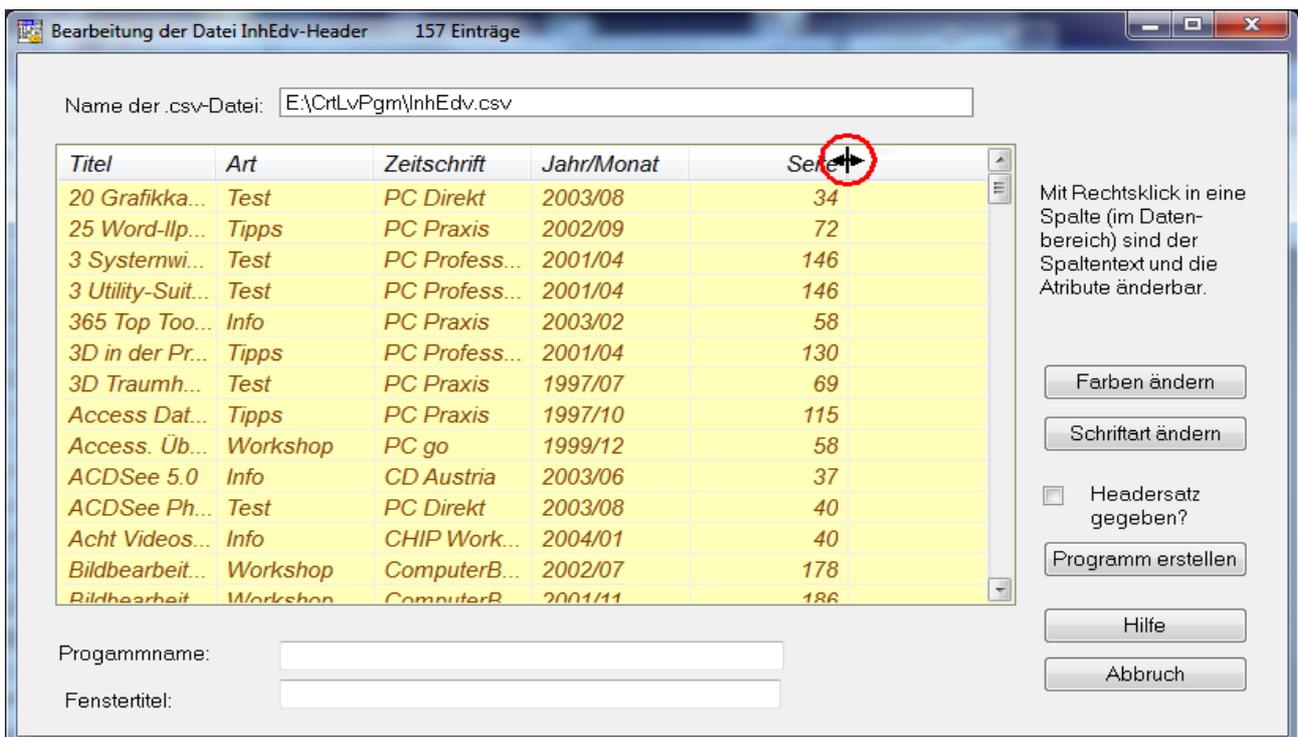


Bild 14.

Für die Schrift habe ich braun gewählt. Das ist natürlich Geschmackssache. Sollte das Ergebnis nicht gefallen, können die Farben erneut eingestellt werden.

Auch kann die Schriftart erneut geändert werden.

Da die Spalten derzeit noch eine einheitliche Breite haben ist es Zeit auch hier etwas zu ändern.

Bewegt man den Cursor über den Spaltenbuttons so ändert sich die Cursoranzeige. In diesem Zustand kann mit gedrückter **Linker Maustaste** die Breite der Spalte geändert werden.

Zweckmäßig beginnt man mit der ganz rechten Spalte. Vergrößert man Spalten kann die Anzeige auch über den rechten Fensterrand hinausgehen. Das macht aber nichts.

Programmbeschreibung Create-Lv-Pgm

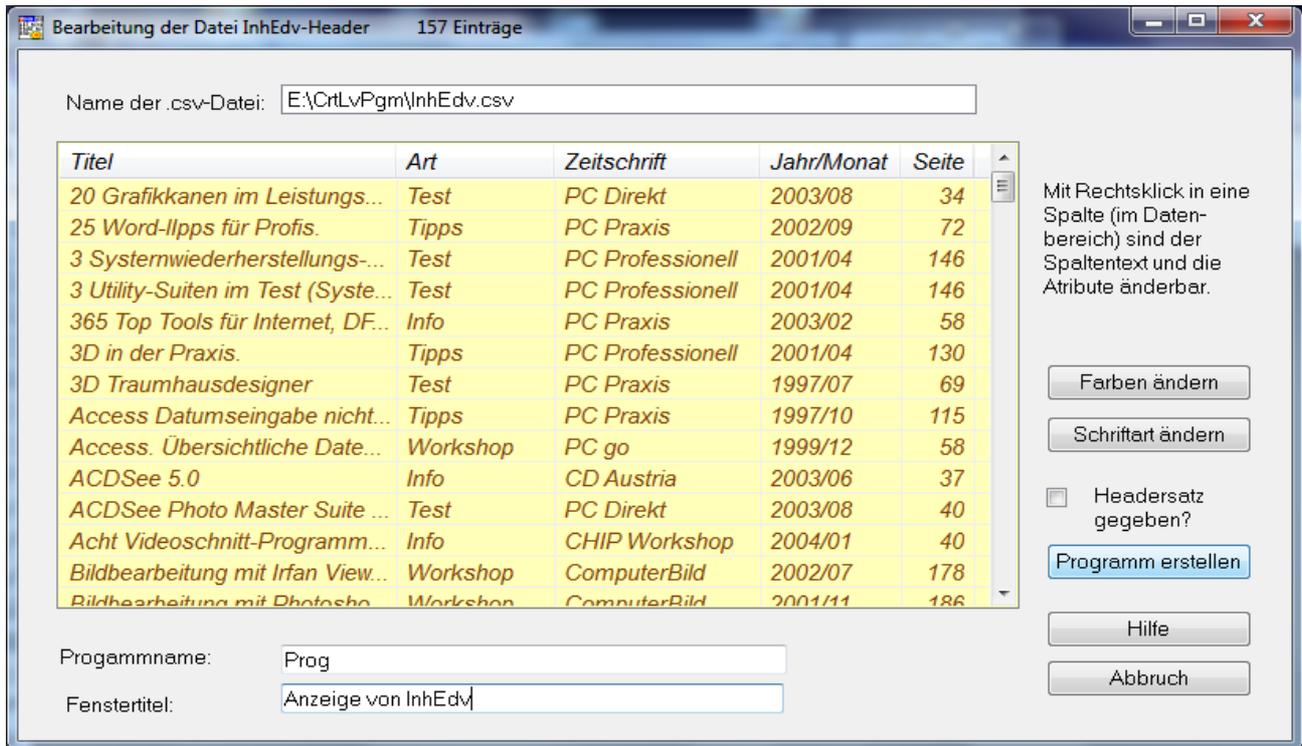


Bild 15.

Hier habe ich alle Spalten auf eine sinnvolle Breite eingestellt. Nur die ganz linke Spalte könnte viel breiter eingestellt werden (was auch problemlos möglich ist).

Wenn jetzt auch noch ein Programmname angegeben und eine Überschrift als Fensterstitel angegeben wird ist nur noch zu beachten ob **Headersatz gegeben?** richtig aktiviert oder nicht aktiviert ist (siehe Bild 04 und Bild 05).

Mit Klick auf den Button **Programm erstellen** wird das Programm (im selben Verzeichnis von **Create-Lv-Pgm**) erstellt. Das Aussehen des ListView-Fensters wird (fast) so aussehen wie es hier im Bild 15 aussieht.

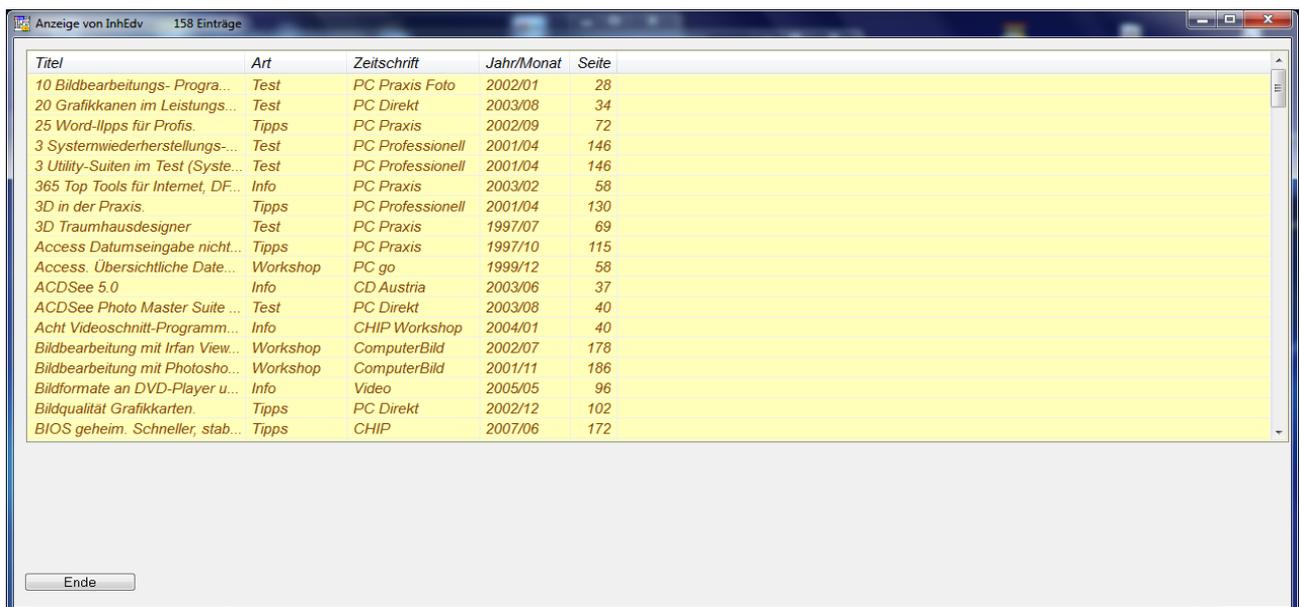


Bild 16.

Im erstellten Programm wurde mit Absicht das Hauptfenster sowie das ListView Fenster auf maximale Breite erstellt!

Programmbeschreibung Create-Lv-Pgm

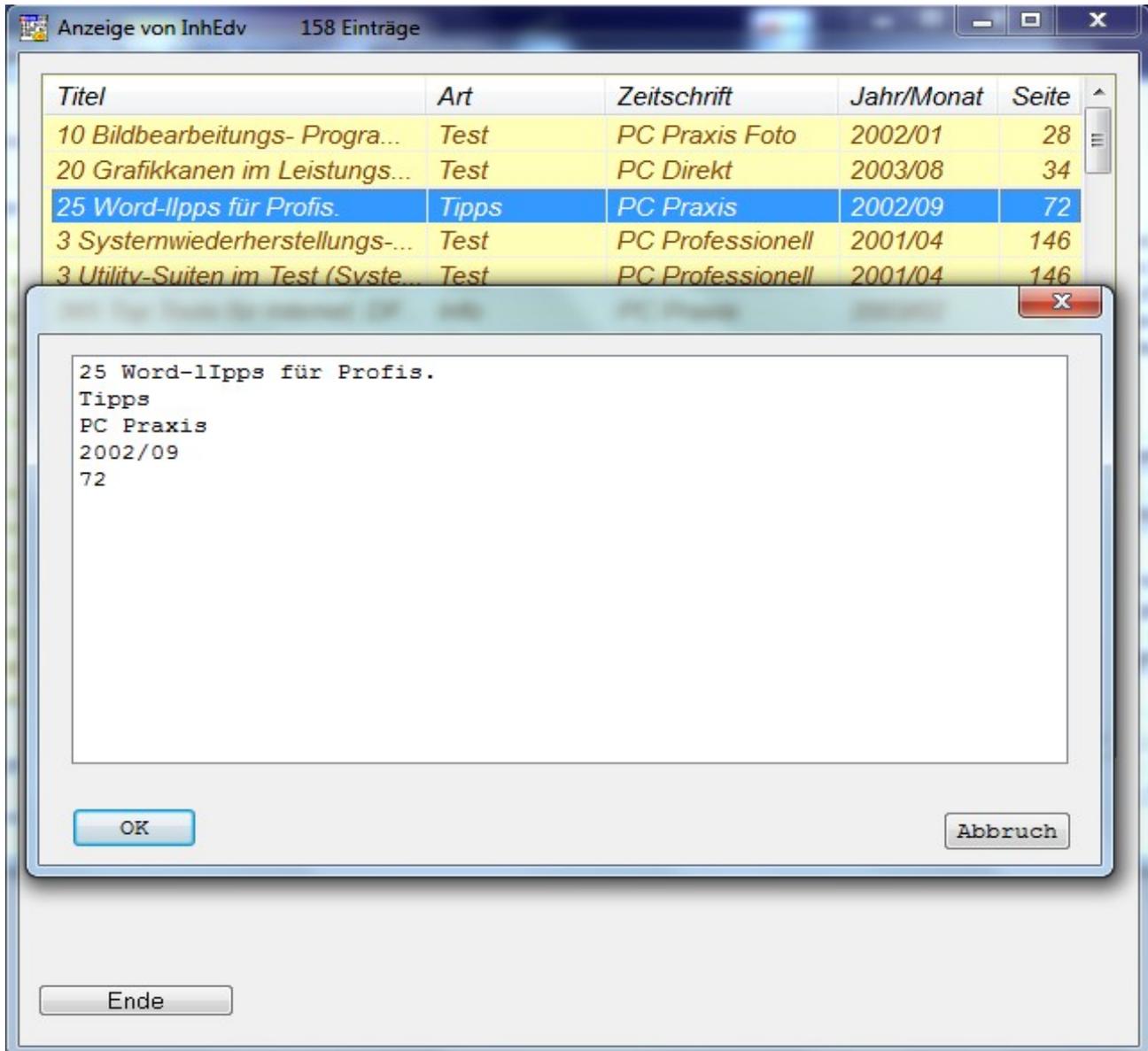


Bild 17.

Wird eine Zeile durch **Doppelklick** ausgewählt können die Daten der Spalten dieser Zeile verarbeitet werden (das Programm wird vermutlich ja nicht nur zum Anzeigen der Datei verwendet werden).

Nur zum ausprobieren wird hier mit dem **Doppelklick** eine Listbox angezeigt in der die Werte der Spalten untereinander angezeigt werden.

Nachbearbeitung des erstellten Programmes

Grundsätzlich sollte das Programm lauffähig sein. Doch kann es sinnvoll sein einige Werte zu ändern.

Das Dialogfenster ist bewusst auf maximale Größe erstellt. Das ListView Fenster ebenfalls.

```
'-----  
$I E:\Dll-Lib\ListView_Funktionen.inc ' <==== ACHTUNG, Pfad anpassen oder ohne Pfad,  
      ' wenn im selben Verzeichnis dieses Programmes.  
  
-----  
  
' == die Breite (hier %MaxX-40) könnte geändert werden. Die erforderliche Breite wäre 600 Pixel!  
ShowListView(listview&,12,12, (%MaxX-40),408)  
  
' == die Breite (hier %MaxX-10) könnte geändert werden. Die erforderliche Breite wäre 640 Pixel!  
Window 0,0-(%MaxX-10),630
```

Bild 18.

Bei (ungefähr) der Zeile 16 `$I E:\Dll-Lib\ListView_Funktionen.inc` ist darauf zu achten, dass hier der Pfad zu der Include Datei stimmt.

Ich empfehle sehr die [ListView.dll](#) von der Seite von **Frank Abbing** herunter zu laden und zu installieren.

Ich habe zwar die [Include-Datei](#) und auch die [ListView.dll](#) dem Paket beigelegt das ist aber nur vorgesehen um die Funktion von testen zu können!

Die nächste Änderung (nicht zwingend!) bei ungefähr der Zeile 118 betrifft die Breite des ListView Fensters, das mit `ShowListView` eingestellt werden kann.

Wurde diese Änderung durchgeführt könnte dann auch die Breite des Hauptfenster verringert werden.

Das ist bei ungefähr der Zeile 57 möglich. In der Anweisung `Window`.

Da bei `InsertColumn` nur der Spaltentext, aber keine Breite und keine Ausrichtung angegeben ist kann hier nur der Text geändert werden(`InsertColumn` wird nur eingesetzt für eine Datei ohne Headersatz!).

Spaltenbreiten können nur in den Anweisungen `SetColumnWidth` , Ausrichtungen in Spalten können nur in den Anweisungen `SetColumnAlignment` geändert werden.

Da das erstellte Programm wahrscheinlich nicht nur zum Ansehen der Datei gedacht ist können Funktionen (inkl. Eingabefelder, Buttons etc.) eingebaut werden. Dabei ist dann der Bereich:

```
' BEISPIEL zum Auslesen der Werte der einzelnen Spalten in Stringvariablen  
' Die Variablen K1$ - ... sind nur beispielhaft  
' K1$ = String$(itemtexts#,Long(nurso#,0))  
' K2$ = String$(itemtexts#,Long(nurso#,4))  
' K3$ = String$(itemtexts#,Long(nurso#,8))  
' . . . .
```

nach Einsetzen der Stringvariablen (anstelle `K1$` ...) zu aktivieren.

Zum Test ist eine Schleife eingebaut. Wird eine Zeile Doppel-geklickt so werden die Inhalte der Spalten einer Zeile in der `Listbox$` angezeigt.

Programmbeschreibung Create-Lv-Pgm

```
' === nur zum Test wird hier die einfach- oder doppelgeklickte Zeile in die Listboxliste  
' übertragen und angezeigt (jede Spalte in eine Zeile der Liste)  
Z& = 0  
ClearList  
WhileLoop 5 ' Anzahl Spalten  
    AddString String$(Texts#,Long(Pointer#,Z&))  
    Z& = (Z& + 4)  
EndWhile  
ListBox$("",2)
```

Dieses Prinzip könnte zum Auslesen der Spaltenwerte aber auch angewendet werden. Damit könnten die einzelnen Werte in eine Liste oder ein Array für die weiter Verwendung ausgelesen werden.

Diese Testschleife ist zu entfernen, wenn Daten der Spalten einer Zeile so wie oben (**K1\$ = ...**) ermittelt werden.

Bemerkungen zur ListView.dll und zu Programm SumatraPDF

Im Programm **Create-Lv-Pgm** ist auch die Prozedur **Suche_den_Pfad** eingebaut. Dies ist ein Extra, welches grundsätzlich nicht notwendig ist wenn sich die externen Teile **ListView.dll** und **SumatraPDF.exe** (+**sumatrapdfprefs.dat**) im selben Verzeichnis wie das Programm **Create-Lv-Pgm** befinden (was nicht unbedingt sinnvoll erscheint weil sich dadurch wahrscheinlich diese beiden Objekte gleichzeitig in mehreren Verzeichnissen befinden).

Sind diese beiden Objekte nicht im Programm-Pfad muss im Programm der Verweis zu den beiden Objekten auch den Pfad beinhalten.

Oder es werden die Pfade zu diesen Objekten mithilfe der Prozedur **Suche_den_Pfad** und der Datei **Profan_Pfade.INI** zur Ausführungszeit ermittelt.

Wird das mit **Create-Lv-Pgm** erstellte Programm als .exe weitergegeben sollte unbedingt die Prozedur **Suche_den_Pfad** eingebaut sein, da hier eine Pfadangabe problematisch ist (aus dem selben Grund wie zuvor beschrieben, wenn sich diese Objekte sinnvollerweise nicht im selben Verzeichnis befinden) weil nicht anzunehmen ist, dass beim Benutzer des erstellten Programmes gleichnamige Verzeichnisse mit den Objekten existieren.

Was macht die Prozedur **Suche_den_Pfad** ?

Die Prozedur sucht das zu findende Objekt (z.B. **ListView.dll**) zuerst im eigenen Verzeichnis. Existiert es hier wird nicht weitergesucht. Existiert es nicht wird das Objekt in der Datei **Profan_Pfade.INI** gesucht. In beiden Fällen ist der Rückgabewert aus der Prozedur der Pfad zum gesuchten Objekt (Laufwerk:\Pfad\Objektname).

Die Datei **Profan_Pfade.INI** ist auf dem Laufwerk im Rootverzeichnis abzulegen auf dem sich auch das Windowsverzeichnis befindet (Normalerweise **C:**).

Ein Beispiel dieser Datei ist beigelegt und ist auf das vorgesehene Verzeichnis zu kopieren. Danach muss mit dem ebenfalls beigelegten Programm **Updat_Pfade.exe** die Datei korrigiert werden damit die Pfade richtiggestellt werden.

In der Datei befinden sich mehrere Einträge. Wichtig sind nur die Verweise zur **ListView.dll** und zu **SumatraPDF.exe** .

Alle anderen Einträge in der Datei **Profan_Pfade.INI** haben keine Bedeutung.

Eine genaue Beschreibung zu **Suche_den_Pfad** kann von <http://www.gerhard-putschalka.xprofan.com/suche-den-pfad.zip> heruntergeladen werden.

Befinden sich sowohl die **ListView.dll** als auch **SumatraPDF.exe** (+ **sumatrapdfprefs.dat**) im selben Verzeichnis wie **Create-Lv-Pgm** bzw. dem damit erstellten Programm so wird die Datei **Profan_Pfade.INI** nicht gebraucht!