12.01.2021

# CSV File Importer

## Dokumentation

Ulrich Stark
OTH AMBERG-WEIDEN

# Inhalt

# 1 Grundlegendes

Das Projekt teilt sich in das „src“ und das „example“ Verzeichnis auf. Im „src“-Verzeichnis befindet sich der Python-Quellcode für den CSV File Importer und einer grafischen Beispielanwendung, die mit dem Framework Tkinter erstellt wurde. Das „example“-Verzeichnis erhält Beispieldateien zum Importieren. Diese teilen sich in CSV, XML und XSL Dateien auf. Alle der Aufgabe beigelegten Dateien wurden beigefügt und wurden um eigene Testdateien ergänzt.

Entwickelt wurde mit dem Versionsverwaltungstool Git. Es steht als Git-Repository unter https://github.com/ulrichstark/csv-importer öffentlich bereit. Die Commit-Historie kann ebenfalls dort nachvollzogen werden.

Dem Projekt liegt außerdem noch diese Dokumentation als Microsoft-Word und PDF-Datei bei. Über das PowerShell Skript „generateDoc.ps1“ kann zu jedem Modul im „src“-Verzeichnis mithilfe von pydoc eine einfache Dokumentation generiert werden.

Um die grafische Beispielanwendung oder das Konsolen-Testprogramm zu starten, müssen zuerst alle benötigten Python Module installiert sein. Dazu zählen die Module pandas, lxml, tkinter, chardet, csv, re und pandastable. Dann sollte mit dem Kommandozeilenbefehl „cd src“ das aktuelle Verzeichnis auf das Quellcode-Verzeichnis geändert werden. Um dann die grafische Beispielanwendung zu starten, reicht der Befehl „python gui.py“. Zum Starten des Konsolen-Testprogramms benutzen Sie bitte den Befehl „python main.py“

# 2 Funktionen

## 2.1 Importieren von CSV-Dateien

Es können CSV-Dateien mit beliebigem Format importiert werden. Dabei erkennt das Programm automatisch das Trenn- und Quotierungszeichen und die Kodierung einer Datei. Außerdem wird versucht, aus der ersten Datenzeile im Vergleich zu den anderen Zeilen zu schließen, ob es sich bei ihr um eine Überschriftenzeile handelt.

Dem Benutzer steht es dabei frei, diese Parameter im Nachhinein zu verändern. Zum Beispiel, wenn das Trennzeichen einer Datei falsch erkannt wurde und somit kein fehlerfreies Importieren gewährleistet werden kann.

## 2.2 Importieren von XML-Dateien

Zusätzlich dazu steht auch der Import von XML-Dateien bereit. Dazu muss eine XSL-Datei (= Extensible Stylesheet Language) angeben werden, die diese zu importierende XML-Datei in das CSV-Format „transformieren" kann.

Auch bei dieser Möglichkeit des Importierens wird die Dateikodierung und die speziellen Parameter des CSV-Formats automatisch erkannt. Die Ausgabe der XSL-Datei kann also eine beliebiges, aber geläufiges Trenn- und Quotierungszeichen wählen und sollte trotzdem ohne Fehler importiert werden können.

## 2.3 Zusammenfügen

Jede importierte Datei (CSV oder XML) wird in eine interne Tabelle im Speicher des Computers zusammengefügt. Dabei spielt die Reihenfolge der importierten Dateien und die Spaltenanzahl eine wichtige Rolle. Der erste Import setzt mit seiner Anzahl der Spalten die benötigte Spaltenanzahl aller folgenden Imports fest. Sollte eine Datei importiert werden, die sich von dieser Anzahl unterscheidet, wird sie übersprungen und ein Fehler wird ausgegeben.

Für die Beschriftung der Spalten wird die erste gültige Überschriftenzeile eines Imports herangezogen. Wenn keine der importierten Dateien eine derartige Kopfzeile besitzt, kann das Programm aus den Zelleninhalten in der Spalte auf den Datentyp schließen und setzt diesen mit einer fortlaufenden Nummer als Beschriftung.

Folgende Datentypen können mithilfe von regulären Ausdrücken erkannt werden: Geo-Koordinaten, E-Mail-Adressen, URLs, Datum kombiniert mit Uhrzeit, Datum, Uhrzeit, Dezimalzahlen, Ganzzahlen oder boolesche Ausdrücke. Sollte keiner dieser Datentypen zutreffen, wird der allgemeine Typ „Text" angenommen.

## 2.4 Exportieren

Die kombinierten importierten Dateien lassen sich daraufhin als CSV- und XML-Datei exportieren. Bei beiden Möglichkeiten müssen der Name, Pfad und Kodierung der zu exportierenden Datei ausgewählt werden.

Während des Exportierens einer CSV-Datei kann zusätzlich dazu noch ein beliebiges Trenn- und Quotierungszeichen gewählt werden. Diese werden zur Generierung der Datei genutzt.

# 3 Grafische Anwendung

Die Grafische Anwendung dient als Beispiel für ein Programm, das die Programmierschnittstellen dieses Projekts benutzt. Entwickelt wurde sie mit dem GUI-Framework Tkinter, das in Python standardmäßig integriert ist.

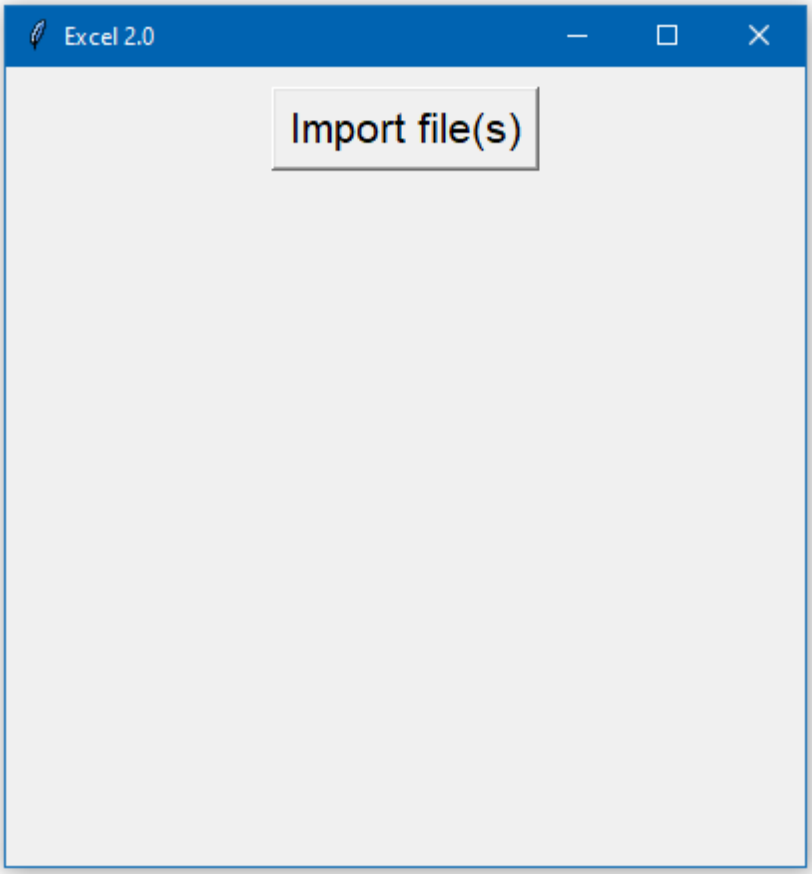


*Abbildung 1: Die Beispielanwendung direkt nach dem Starten*

Nach dem Starten wird der Benutzer von einem einfachen Fenster auf seinem Desktop empfangen, das einen Knopf mit der Aufschrift „Import file(s)“ enthält. Die restlichen Bedienelemente werden bewusst erst eingeblendet, nachdem der Nutzer Dateien zum Importieren ausgewählt hat. Dadurch wird er bei Programmstart nicht von unnötigen Eingabefeldern und Knöpfen abgelenkt.

Wenn er den Knopf zum Importieren der Dateien klickt, erscheint der vom Betriebssystem bereitgestellte Dateiauswahl-Dialog.

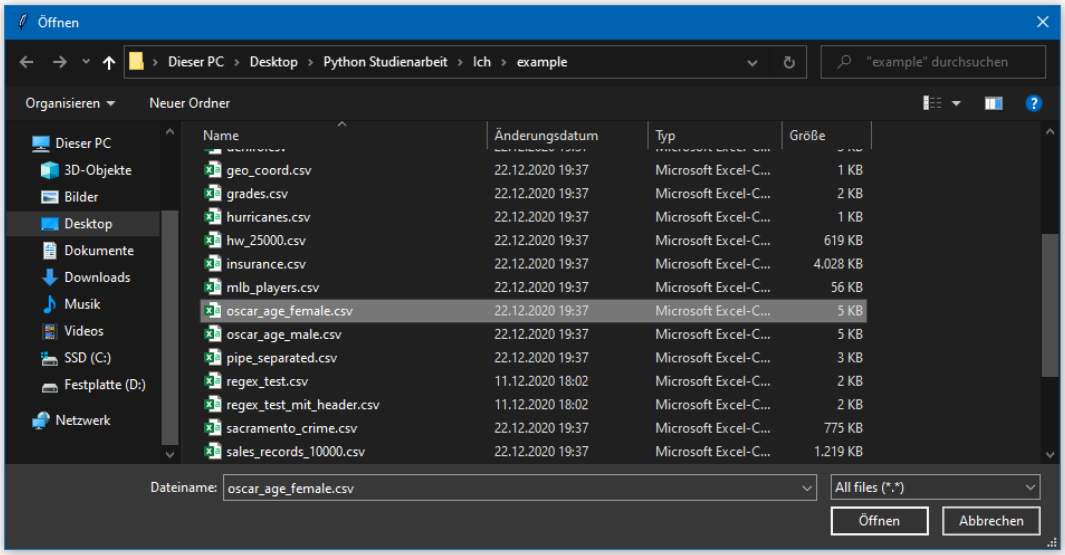


*Abbildung 2: Der Dateiauswahl-Dialog zum Importieren der CSV- und XML-Dateien*

Dieser Dialog wurde gezielt so konfiguriert, dass er eine Mehrfachauswahl an Dateien zulässt. Außerdem wird standardmäßig das Verzeichnis mit den Beispieldateien geöffnet. Diese beiden Faktoren steigern die Effizienz beim Benutzen und Testen der Beispielanwendung.

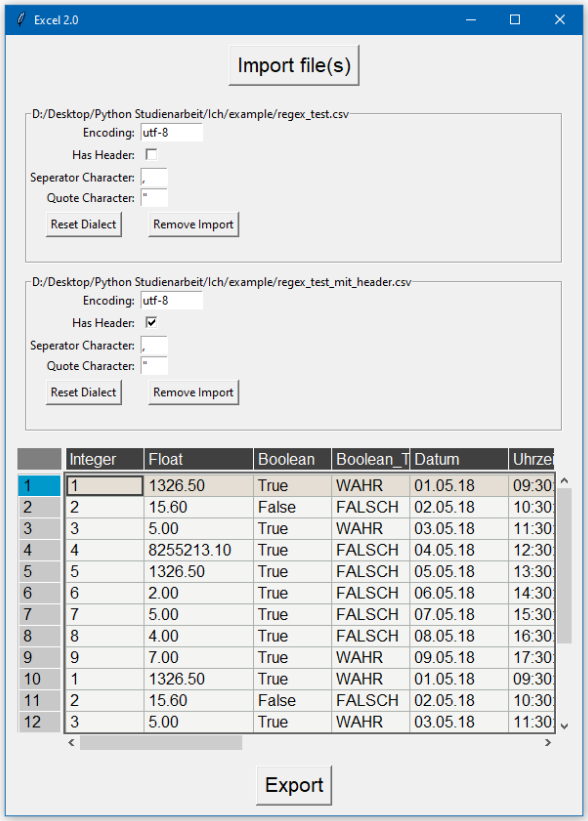


*Abbildung 3: Ansicht der Beispielanwendung nach dem Importieren von zwei Dateien*

Nachdem mindestens eine Datei importiert wurde, erscheinen die restlichen Bedienelemente. Das Fenster teilt sich nun in vier Bereiche.

Erstens in den Knopf zum Importieren. Darauffolgend ein Rahmen („Frame“) für jede importierte Datei. Drittens eine Tabelle, die als Vorschau für die zusammengefügten Dateien dienen soll. Und zuletzt ein Knopf mit der Aufschrift „Export“. Ein Klick auf diesen öffnet das Fenster zum Exportieren.

## 3.1 Ansicht der importierten Dateien

In diesem Teil der Anwendung erhält jede importierte Datei einen Bereich abgegrenzt durch einen Rahmen („Frame"), in dem ihre zugehörigen Parameter angepasst oder zurückgesetzt werden können. Außerdem steht ein Knopf zum Entfernen dieses Imports bereit.

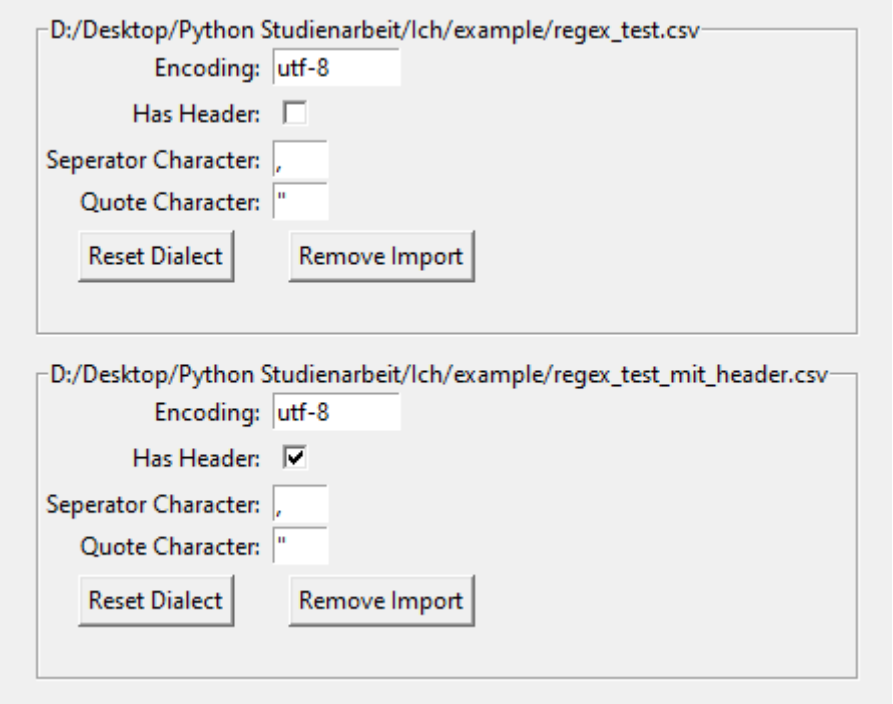


*Abbildung 4: Zwei importierte CSV-Dateien mit jeweils Parameter und Aktionen*

Der Frame für eine importierte CSV-Datei ermöglicht die Anpassung der Dateikodierung, des Trenn- und Quotierungszeichens und ob die Datei eine Kopfzeile besitzt. Diese Werte werden benutzt, um die Datei zu importieren und werden in den meisten Fällen automatisch richtig detektiert bevor der Nutzer eine Eingabe machen muss. Mit dem „Reset Dialect"-Knopf werden die Änderungen des Benutzers verworfen und die Werte abermals aus der Datei detektiert.

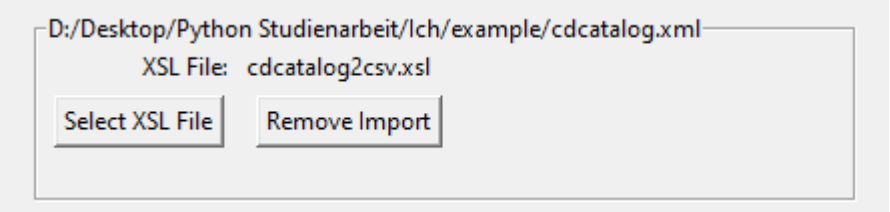


*Abbildung 5: Importierte XML-Datei mit ausgewählter XSL-Datei*

Im Frame einer importierten XML-Datei steht zusätzlich zu dem Entfernen-Knopf nur ein Knopf zum Auswählen einer XSl-Datei bereit.

Diese wird benutzt, um den Inhalt der XML-Datei in das CSV-Format zu überführen.

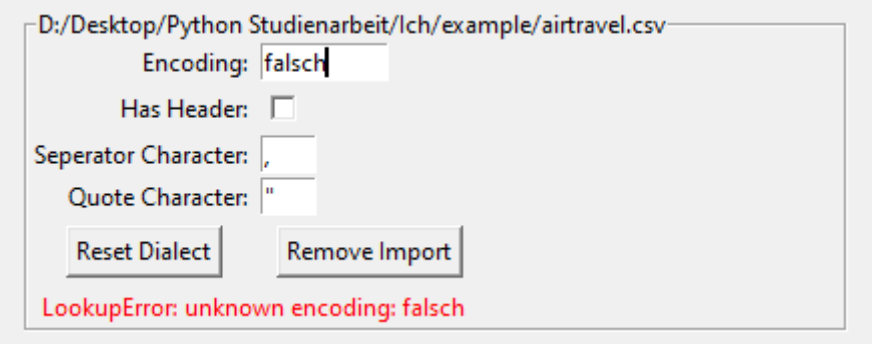


*Abbildung 6: Fehlermeldung bei falscher Angabe der Dateikodierung*

Wenn beim Importieren einer Datei ein Fehler auftritt oder die Parameter ungültig sind, wird ein Fehler in dem Frame ausgegeben.

## 3.2 Vorschau-Tabelle

## 3.3 Fenster zum Exportieren

# 4 Programmierschnittstellen