

Serielles Bearbeiten von Dateien

Vom 'aufwändigen' Befehl für eine einzelne Datei zum automatischen Bearbeiten beliebig vieler Dateien 'per Knopfdruck', am Beispiel des `#*import`-Makros

ITUG Jour fixe am 26.5.2021

Hans-Werner Bartz (Hans-Werner.Bartz[at]adwmainz.de)

1. Befehl in der Kommandozeile

```
Gib Kommando >#*import,datei01.rtf,datei01.tus,ign=-,lo=+
```

2. Ausführen der 'Programmdatei' *import.p*

```
***** | ——— Dateianfang ———  
1      | #*import,datei01.rtf,datei01.tus,ign=-,lo=+  
***** | ——— Dateiende ———
```

```
Gib Kommando >#t,import.p
```

3. flexible Variante mit Parameter

```
***** | ——— Dateianfang ———  
1      | #*import,?1.rtf,?1.tus,ign=-,lo=+  
***** | ——— Dateiende ———
```

```
Gib Kommando >#t,import.p,pa=datei01
```

```
Gib Kommando >#t,import.p,pa=datei02
```

```
...
```

- Beim Ausführen von `#tue` können bis zu 9 Parameter (getrennt durch ' ') angegeben werden, die die in der Programmdatei die mit `?1` - `?9` benannten Platzhalter ersetzen

- flexible Variante (vollständig mit `#an`, `#da`, `#ab`)

```
***** | ——— Dateianfang ———  
1      | #an?1.rtf  
2      | #da,?1.tus,seq-ap  
3      | #*import,?1.rtf,?1.tus,ign=-,lo=+  
4      | #ab,?1.rtf'?1.tus  
***** | ——— Dateiende ———
```

- soll die Zieldatei einen **anderen Namen** erhalten, muß bei den Angaben zur Zieldatei ?1 zu ?2 werden:

```
***** | ——— Dateianfang ———
1      | #an?1.rtf
2      | #da,?2.tus,seq-ap
3      | #*import,?1.rtf,?2.tus,ign=-,lo=+
4      | #ab,?1.rtf'?2.tus
***** | ——— Dateiode ———
```

Gib Kommando >#t,import.p,pa=datei01'neu1

4. Ausweitung: beliebig viele Dateien

4.1. Schritt 1: Ermitteln der Dateien (Dateiliste in vorher erstellte Datei sc1 ausgeben)

Gib Kommando >#li,da,po=datei0*.rtf,da=sc1,lo=+

Gib Kommando >#liste,MODUS=datei,POSITIV=datei0*.rtf,DATEI=sc1,LOESCHEN=+

Gib Kommando >#li,da,sc1,+,,,datei0*.rtf

Spezifikation | auswählbare Option | frei wählbar

- Spezifikationen können abgekürzt werden, solange sie eindeutig bleiben
- Reihenfolge der Spezifikationen ist beliebig, wenn der Spezifikationsname genannt wird (2. Zeile)
- ohne Spezifikationsname gilt die im Handbuch beschriebene Reihenfolge (3. Zeile)
- Kombinationen sind möglich (1. Zeile)

4.2. Schritt 2: Kombination mit Schritt 3.2.

Gib Anweisung >a,,,*|.rtf|#t,import.prg,pa={+1=} |

```
***** | ——— Dateianfang ———
1      | JF*DATEI01.RTF
.....
1      | #t,import.p,pa=JF*DATEI01
```

- Variante: Erstellen einer Gesamtdatei

Gib Anweisung >a,,,*|#*import,{=0=},gesamt.tus,ign=-,lo=-

(Achtung: falls dieses Programm mehr als einmal für dieselben Dateien genutzt werden soll, dann muß vorher der alte Inhalt der Zieldatei *gesamt.tus* gelöscht werden)

5. Zusammenführen von 4.1 und 4.2 in der Datei *import.prg*

```
*****  ——— Dateianfang ———
 1      #da,sc1'sc2'pr1,fr=-
 2      #li,da,po=datei0*.rtf,dat=sc1
 3
 4      #ko,sc1,sc2,+,+,*
 5      xx          |*.rtf|#t,pr1,pa={+1=}|
 6      *eof
 7
 8      #um,*,pr1,lo=+
 9      #an,?1.rtf
10      #da,?1.tus,seq-ap
11      #*import,?1.rtf,?1.tus,ign=-,lo=+
12      #ab,?1.rtf'?1.tus
13      *eof
14
15      #t,sc2
*****  ——— Dateieinde ———
```

Gib Kommando >#t,import.prg

6. Erstellen einer speziellen „Import-Sitzung“

(diese Sitzung soll/kann nur RTF-Dateien in gleichnamige Tustepdateien umwandeln)

1. neues Verzeichnis mit eigener Sitzung (via #*desi, mit icon!) anlegen
2. in der Datei *import.prg* (aus Nr. 5) folgendes ergänzen:
 - am Dateianfang: #pr, frei
 - am Dateieinde: #no,+
3. *import.prg* in *tustep.ini* umbenennen und in das neue Verzeichnis kopieren

Anhang: Umgang mit ‚illegalem‘

```
***** | —— Dateianfang ——  
1      | #da,pr1'gesamt.tus,seq-ap  
2      | #ma  
3      | $$ mode tuscript, {}  
4      |   set dat_dir = "0:\goethe\Korrektur\biographica\RA\01\regesta_and_transcriptions"  
5      |  
6      |   dat_list = file_names (-, "{dat_dir}")  
7      |  
8      | file/erase/program "pr1"  
9      |   data #de,da=xyz:{dat_dir}\{dat_list} #an,xyz,tr=- #um,xyz,gesamt.tus,+,- #ab,xyz  
10     | endfile  
11     |  
12     | *eof  
13     |  
14     | #t,pr1  
***** | —— Dateiende ——
```

Beispiel für das Erstellen einer Gesamtdatei aus beliebig vielen Einzeldateien ohne Berücksichtigung, ob diese ‚legale‘ oder ‚illegale‘ Dateinamen haben.

- entscheidend ist hierfür die Funktion `file_names` (Z. 6) aus `#makro`
- das Ergebnis der Funktion wird in die vorher (1) erstellte Datei `pr1` geschrieben (8-10)
- gleichzeitig werden alle Kommandos, die für die Weiterverarbeitung nötig sind, für jeden Dateinamen ergänzt:
`#de,da=xyz:{dat_dir}\{dat_list}` = Definieren eines alias `xyz` mit dem vollständigen Pfad der Datei