

python_data_matcher.py



LUZIE U. WINGEN

python_data_matcher.py is a **python** script to match or join the content of two spread sheets by their first column items. The initial purpose was to join marker information from different lists, but it has been proven useful beyond that.

Basic instructions:

Install Python 3.5 or higher (<https://www.python.org>).

Unzip **python_data_matcher.zip**

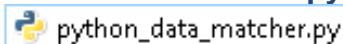
Unzip the zip file on your computer and navigate into the directory **python_data_matcher**, which contains the script **python_data_matcher.py**.

Save spread sheets from Excel as "csv".

The script works on comma-separated spread sheets or "csv" files. Any Excel spread sheet needs to be saved as "csv" first. "csv"-files are best put into the directory with the **python** script - less typing of directory names.

Start the **python_data_matcher.py** script

On Windows click on **python_data_matcher.py** icon to execute the script.



A window will pop up and ask for the name of the first file.

```
C:\Program Files\Python2.7\python.exe
#####
#                                     ##
#           Python Data Matcher          ##
#                                     ##
#   Joining two 'csv' spread sheets by matching the items in the first columns. ##
#                                     ##
#   The script will ask for the names of two spread sheet files to match.      ##
#   The files have to be in 'csv' (comma seperated value) format.              ##
#   Please type valid file names otherwise the script will fail.                ##
#                                     ##
#   Python Data Matcher written by Luzie U. Wingen, JIC, 8 November 2018.      ##
#                                     ##
#####
#                                     ##
#   Please type the name of the fist csv' spread sheet:                        ##
#                                     ##
#   _
```

Type or copy the name of the first file and hit "return".

Be aware: **python** is case-sensitive!

Also: The file extension ".csv" is part of the file name.

```
#####
##                                     ##
## Please type the name of the first csv' spread sheet:                       ##
##                                     ##
## data/UK_winter_wheat_RL_2017-18.csv                                       ##
## Load data from: data/UK_winter_wheat_RL_2017-18.csv                       ##
## 37 rows found in file.                                                    ##
##                                     ##
## length column headers:', '6', '; length row 1:', '6')                    ##
#####
##                                     ##
## Please type the name of the second 'csv' spread sheet:                     ##
##                                     ##
## _
```

The script will then ask for the name of the second file.

Type or copy the name of the second file and hit "return".

The script reports back if it finds items in the first column of the second file that are not present in the first column of the first file.

Adding items from the second file to the first

If you want items added answer with: "y" + "return".

If you don't want items added just hit "return".

```
## data/UK_winter_wheat_RL_2018-19.csv
## Load data from: data/UK_winter_wheat_RL_2018-19.csv
## 42 rows found in file.
##                                     ##
## length column headers:', '7', '; length row 1:', '7')                    ##
#####
##                                     ##
## Python Data Matcher will now merge the spread sheets:                     ##
##                                     ##
#####
The following entries in file 2 were not found in file 1 (only first 20 given):
['KWS Zyatt', 'Crusoe', 'Elicit', 'KWS Jackal', 'Elation', 'Bennington', 'LG Sundance',
wn', 'Moulton', 'RGT Gravity', 'Gleam', 'KWS Kerrin', 'Shabras', 'KWS Santiago', 'Freis
#####
##                                     ##
## ('## ', '19', ' item(s) in the second file are not present in the first file.') ##
##                                     ##
## Do you want these items added to the merged table?                         ##
##                                     ##
## Type 'y' for yes (return or any other key for no):                         ##
## _
```

python_data_matcher.py produces a new file with the joint items.
As a filename it will use the first file name with an added '_extended'.

All done

Hit "return" to leave the script and inspect the new file which will be in the **python_data_matcher** directory.

14/11/2019