Database Merge & Compare

This program compares many customer databases against a primary database, finding, sorting, and extracting the most valuable data on these matching customers.

This document details the general logic of the program and its programmable settings.

This program has been made through private agreement, with permission to share for the author's portfolio purposes only. Use of this software for commercial or private purposes is expressly forbidden.

Sections

Program Logic

To ensure smooth use of the program, the following section is provided as reference should you encounter errors or unexpected behaviour.

Load Settings

The program begins by automatically loading from the settings.txt file in the program folder. If it is not found, it will restore default from the backup settings.bak file. If there are any errors in the settings.txt file, i.e., settings keys changed or setting values input incorrectly, it will inform the user of corruption and prompt to restore from defaults. Once loaded, the key settings are displayed to the user.

If fast mode has been enabled, the program will continue on to load files. Otherwise, it will display the settings and prompt the user to change these, thereafter doing the same for column names.

Check Directories and Files

If the directories specified in the settings.txt are not found, the program will create them, ask the user to add files to the input directory and restart the program.

The program will then check for the presence of the primary database according to the name given in the settings.txt file. If it is not found, the program prompts the user to type its filename.

> **NOTE**: Files moved into the folder once the program has started are not loaded. To load these files, you will need to restart the program.

It then converts any Excel files to csv for fasting loading into the program, moving all remaining Excel files in the input folder to the redundant folder.

> **NOTE**: It is advised to let the program convert Excel files to csv itself. If using your own csv files, please ensure they are ';' delimited.

Load Databases

The program first checks for header rows, identifying number of rows to skip to find the column name row.

> **NOTE**: The program is able to detect header rows if the header rows take up no more than 20 rows. Please ensure the column names rows starts within the first 20 rows of the file before conversion in the previous step.

Data is the loaded. If the user has decided to drop customers who do not wish to be contacted from the data, these customers are removed from the database here. Copying the dataframe and storing the original for later reference for the export, it cleans the following column types in the copied dataframe:

- **Name Columns**: Make all lower case, remove spaces, and any non-alphabetical characters.
- **Phone Columns**: Remove non-digit characters and drop prefixes defined by Phone Drop setting.
- **Email Columns**: Make all lower case and remove spaces.
- **Phone Columns**: Remove non-digit characters.

> **NOTE**: The program only processes the columns it can find in the dataframe as defined by the column names settings.

Matching

Match by the following logic:

1. Match where names columns are identical.
2. Check options file for loop columns. If only match by names enabled, stop.
3. Get column names for first matching condition, i.e., phone numbers.
4. Loop through each available customer column name from the settings and match to all primary database columns for that matching condition. If a column is not found in the customer database, skip it. Extract their IDs
5. Loop back to 3 for each other specified match conditions.
6. Drop any IDs that are perfect matches from the uncertain matches.
7. Drop any IDs in the other conditions from the missing data matches.
8. Match uncertain or missing data matches by medical area. Check the column names in the primary database, and their translations stored in the areas.txt file. These checks occur similarly to the method in the previous steps.

Resolve Conflicts: Excel

If the user has opted to resolve by Excel in settings, the program exports the matches from the copied dataframe (containing only the processed information) for each of the 3 cases. It saves this file in the defined check folder with the same filename as the original customer database with suffix '_check'. This file is automatically opened for the user.

The user must then decide which IDs to remove and save the file. The program will then re-import this Excel file, reading the remaining IDs, adding those to the final match list.

> **NOTE**: The user must save the file after deleting rows, or the program may keep *ALL* conflicts.

Resolve Conflicts: Program

Otherwise, if the user has instead opted to resolve conflicts within the program in settings, they are shown the conflicting IDs. They are then given the option to view more information on these conflicts in the program window by inputting in IDs. The following options are possible:

- Enter numbers separated by a space e.g. '2 8 9 1000 92857'
- Enter 'all' to view information on all conflicts
- Enter 'n' to stop viewing and specify the IDs they would like to keep

After entering 'n', the user must then enter the IDs they would like to keep with the following options possible:

- Enter numbers separated by a space e.g. '2 8 9 1000 92857'
- Enter 'all' to keep all conflicts
- Enter 'none' to drop all conflicts

The program then shows the number of conflicts it is keeping and continues.

Export Reduced Sheets

The first export saves the matched raw information from the given databases to the final folder. The information exported can be defined settings.

- Program extracts the IDs and customer indices from the match data and gets only those entries from the original database
- If the user has opted to export customer data, get all customer data. Otherwise, just get the data from the columns used for matching from customer database.
- Add customer data to a separate sheet if the user has opted to. Otherwise, append to the primary data with a spaced column between the two containing 'CUSTOMER DATA >>>' to separate the data.
- Save to the specified final folder with the same filename as the original customer database with suffix '_final'

Export Mailing List

If specified in the settings, the program will further process the data into a mailing list sheet appended to the final file. This list is processed as such:

- Get all contact information found from both databases as defined in the column settings
- Merge duplicate entries, keeping only unique contact data on each person from each database.
- Sort emails according to the Email Sort setting.
  - Email prefixes given in this list are de-prioritised, i.e., emails not beginning with the prefixes in this list are put first.
    - **NOTE**: These de-prioritised emails are sorted in the same order as given in the settings and can therefore be used to sort emails by lowest priority.
- Sort phone numbers according to the Phone Sort and Phone Drop settings.
  - Check for identical phone numbers by dropping prefixes found in the Phone Drop setting, comparing for each entry. Keep numbers containing prefixes.
  - Sort numbers according to the prefixes in the Phone Sort setting. Numbers in this list are prioritised, i.e., numbers beginning with the prefixes in this list are put first.
    - **NOTE**: Numbers sorted in the same order as given in the settings e.g., if [01, 02] given, all numbers beginning with 01 appear first, 02 in the middle, with the rest at the end.
- If the given greeting is also in the Email Add setting, fix typo in greeting:
  - 'geeherte' -> 'geehrte' *AND* 'geeherter' -> 'geehrter'
- If the given title column is also in the Email Add setting:
  - Get titles from both databases
  - Drop titles in Title Drop setting (case-insensitive)
  - Return list of unique titles in Title column
    - **NOTE**: The order of these titles will be in the same order as given i.e., titles in primary database followed by titles in customer database.

Entries are then split into two sheets based on whether the entries contain email addresses or not. The program then appends these lists to the final export. If enabled in the settings, the program will then repeat the above process for uncertain matches still contained in the intermediary Excel sheet.

Loopback and Finalise

The program loops back to the matching stage for every other customer database found in the file. Once complete it opens the final folder waits for the user to close the window.

Logging

Program exports log files to 'log' folder found in the program directory. These contain further information about each run and can be used for debugging or error correction purposes. If more than 30 logs are in the folder, logs older than 2 months are deleted.

Settings

The settings.txt file found in the program directory controls the default settings, column names, and directories of the program. You may edit this file to change the default behaviour of the program

> **WARNING**: enter settings on one line, do not press return at the end of a line!

> **WARNING**: Do not change the text before : on any line that doesn't begin with #

Lines beginning with # are ignored by the program, feel free to add any of your own comments in this way.

> **WARNING**: Adding lines without # will cause the program to halt.

Program Options

To change program options, please input one or many of the following characters:

> **NOTE**: match conditions may be combined i.e., typing 'ef' matches by name, confirming with email or fax

Enter all program options as a single word (case-insensitive) *e.g., 'pec' or 'ml' (without quotation marks)*

Match Options

- **n**: match by name only (enabled by default in the program)
- **p**: match by phone number
- **e**: match by email
- **f**: match by fax
- **d**: attempt to drop duplicate entries from customer database based on phone, email, and fax number columns
- **z**: if 'd' enabled, also attempt to drop duplicates from primary database
- **i**: drop customer's who do not want to be contacted *(i.e., where Interesse MAFO = 2)*

Export Options

- **x**: export intermediary match cases to Excel spreadsheets
- **l**: append mailing list sheet
- **c**: append all customer's data in final export to primary sheet
- **m**: append matching customer's data to the primary data sheet in final export to primary sheet
- **a**: add customer data as an extra sheet rather than to the primary sheet *(data included can be modified with c or m as described above)*
- **u**: add extra mailing sheet for uncertain matches

Runtime Options

- **s**: fast mode - skip all settings checks and just use the settings in this file
- **v**: view mode - view conflicting entries in program and resolve by manual ID input instead of by external Excel sheet.
  > **NOTE**: This setting is only considered if fast mode is enabled
- **q**: auto mode - skip all user input and automatically run the program *(enables fast mode by default)*
  > **NOTE**: If user has any files open that the program needs to write to, the program now saves them with '_#' suffix where # is an incrementing number.
- **o**: omit mode - when auto mode is enabled, only keep perfect matches
  > **NOTE**: This setting is only considered if auto mode is enabled

Defaults

> - **Options**: npedi xlcau sqo

Column Names

These settings control the column names the program will use to search for matches based on the above settings.

Enter column names as a comma-separated list (case-sensitive)
e.g., 'Telefon-Festnetz, Telefon-Mobil (beruflich), Telefon-Mobil (privat)' (without quotation marks)

Primary Database

The column names for the primary database must be exact and in order.

Defaults

> - **Pri Names** : Vorname, Nachname
> - **Pri Phone** : Telefon-Festnetz (beruflich), Telefon-Festnetz (privat), Telefon-Mobil (beruflich), Telefon-Mobil (privat)
> - **Pri Email** : E-Mail (beruflich), E-Mail
> - **Pri Fax** : Faxnummer (beruflich), Faxnummer (privat)
> - **ID Col** : TID
> - **Interest** : Interesse MAFO
> - **Greeting** : Briefanrede
> - **Title** : Titel

Customer Database

Unlike the primary database, the customer column names may be a list of options of all the possible names given to that data type. For example, if one customer database contains only the column name 'eMail' for emails, but another contains both 'Email' and 'EMail', you may add all three names, and the program will match all the columns it is able to find.

> **WARNING**: While you may define many column names for the customer names setting, please ensure these are in first name to last name order.
>
> - i.e., you may write 'First_Name, Vorname, Firstname, Last_Name, Nachname' but not 'First_Name, Last_Name, Vorname, Nachname, Firstname'

Defaults

> - **Cus Names** : Vorname, First_Name, FirstName, Nachname, Last_Name, LastName, Person Name
> - **Cus Phone** : Phone, Telefon, Phone Nummer
> - **Cus Email** : eMail, Email, EMail, E-Mail
> - **Cus Fax** : Fax
> - **Cus Title** : Titel, Anrede und Titel
> - **Cus Areas** : Person Fachgebiet 1, Person Fachgebiet 2

Mailing List Options

Mailing lists are built from the email column names given above.

> **NOTE**: To use the mailing list function, ensure that primary and customer databases have unique column names for emails!

Enter these options as a comma-separated list (case-sensitive)
e.g., 'sekretariat@, info@, mail@, praxis@, kontakt@' (without quotation marks)

Email Sort

- Emails containing these prefixes will be de-prioritised in the mailing list sheet. Program sorts emails based on order of this list, prioritising emails not in the list. Hence, this can also be used as a pseudo sorting function.
  > **NOTE**: list items must be email prefixes and must end with @ i.e., 'info@'.

Email Add

- Extra columns from the primary database to include in the mailing list export
  > **NOTE**: ID, names, and email columns included by default, do not add these here.

Phone Sort

- Numbers containing these prefixes will be prioritised in the mailing list sheet.Program sorts numbers based on order of this list, de-prioritising numbers not in the list.
  > **NOTE**: list items must be phone prefixes beginning with 0.

Phone Drop

- Phone prefixes/area codes the program will drop before attempting to match by phone number.
  > **NOTE**: must start with 0, or the program may incorrectly match data.

Title Drop

- Titles to drop from the mailing list title column (case-insensitive)

Defaults

> - **Mail Sort** : sekretariat@, info@, mail@, praxis@, kontakt@
> - **Mail Add** : Briefanrede, Anrede, Titel
> - **Phone Sort** : 01
> - **Phone Drop** : 030, 033
> - **Title Drop** : med, Herrn, Frau, Herr, meed

Directories

These settings control the default folders & files from which the program loads and saves files

Enter the exact name of folders/files (case-sensitive)

Folder Name

- The folder name placed on the desktop for files, inside which the following folders are stored

Input

- The input database files to be compared

Check

- Where the Excel sheets for intermediary match cases are saved i.e., Perfect, Uncertain, and Missing Data matches

Final

- Where the final exports are saved to

Primary Data

- Primary database file name to search for in input folder (extension not required)

Defaults

> - **Folder Name** : Database Comparison
> - **Input** : input
> - **Check** : check
> - **Final** : final
> - **Primary Data** : primary_database

Areas

During the check medical areas matching that happens in step 8 of the matching process, the program pulls from a list of alternative names given in the 'areas.txt' file in the program root. These can be translations, or alternative versions of column names you would like to search for in the customer database's medical area columns. The program will consider any entries with matching information a perfect match.

> - **WARNING**: Column names and their translations are case-sensitive e.g., if you wish to add Physiotherapie as an alternative in lower and capitalised case, you must add both 'physiotherapy' and 'Physiotherapy'

The document must be laid out as below *(spaces before and after the colon are ignored)*
i.e., Original Column name, colon (:), list of alternatives separated by a comma.

Example

> **Physiotherapie** : physical therapy, physiotherapy, Physiotherapy
> **Neurologie** : neurology
> **Nuklearmedizin** : Nuclear medicine, radiology

Author Information, Licence, and Acknowledgements

> - **Program Name**: Database Merge & Compare
> - **Version**: 1.0
> - **Date**: 18/06/2021
> - **Author**: George M. Marino

Copyright (C) 2021 George Martin Marino g.marino94@live.com

This file is part of the Database Merge & Compare project.

The Database Merge & Compare project can not be copied and/or distributed without the express permission of George Martin Marino g.marino94@live.com.

This program has been made through private agreement, with permission to share for the author's portfolio purposes only. Use of this software for commercial or private purposes is expressly forbidden.

Xslx2csv (v0.7.8) package obtained from Dilshod Temirkhodjaev from their GitHub account under GNU General Public License as published by the Free Software Foundation. License permits use, redistribution, and modification.