CSV Take Home Reconciliation Challenge

Background:Our primary product involves reconciliation of data, typically represented in CSV files. Reconciliation is the process of ensuring two sets of records are in agreement. Discrepancies or mismatches can occur due to various reasons like missing records, altered data, or addition of new records.

Objective:Your challenge is to create a tool that will read in two CSV files (termed "source" and "target"), reconcile the records, and produce a report detailing the differences between the two.

Requirements:

Input:

- The tool should accept two CSV files as input.
- Each CSV file will have a header row.
- Assume that the first column of each CSV is a unique identifier for the records.
- Processing:
    - Identify records that are present in the source but missing in the target (and vice versa).
    - For records that exist in both files, compare each field to identify discrepancies.
    - Implement a way to handle potential data transformation issues (e.g., date formats, case sensitivity, leading/trailing spaces).

Output:

- Produce a reconciliation report with the following sections:
    - Records present in source but missing in target.
    - Records present in target but missing in source.
    - Records with field discrepancies, highlighting the specific fields that differ.
- The output can be another CSV, a nicely formatted HTML page, or any other easily readable format you choose.
- The backend language used must be Python

Additional Features (Optional):

- Implement a graphical user interface (GUI) for easy file selection and result visualization.
- Allow the user to configure which columns to compare, in case some columns should be ignored during reconciliation.
- Implement fuzzy matching for non-identical but similar records.

Guidelines:

- Code quality matters. Please ensure your solution is well-organized, commented, and tested.
- The challenge is expected to take 3-5 hours, but use as much or as little time as you need to deliver a quality solution.
- Provide documentation or instructions on how to run your tool.
- Scale over millions of rows.

Bonus?:

- While not required, any additional features, insights, or suggestions related to our product will be highly appreciated. This will be required if you are interviewing for **Senior Software Engineer** or a **Tech Lead**
- CI/CD

Further Details:

Sample Input Data

source.csv

ID,Name,Date,Amount
001,John Doe,2023-01-01,100.00
002,Jane Smith,2023-01-02,200.50
003,Robert Brown,2023-01-03,300.75

target.csv

ID,Name,Date,Amount
001,John Doe,2023-01-01,100.00
002,Jane Smith,2023-01-04,200.50
004,Emily White,2023-01-05,400.90
Expected Output Data

reconciliation_report.csv

Type,Record Identifier,Field,Source Value,Target Value
Missing in Target,003,,,,
Missing in Source,,004,,,
Field Discrepancy,002,Date,2023-01-02,2023-01-04

CLI (Command Line Interface) Specifications:

The CLI tool can be named csv_reconciler. Here's a hypothetical usage:

bash

$ csv_reconciler -s source.csv -t target.csv -o reconciliation_report.csv
Where:

-s or --source is the path to the source CSV file.
-t or --target is the path to the target CSV file.
-o or --output is the path to save the output reconciliation report.
Once run, the CLI can provide a summary:

bash

Reconciliation completed:
- Records missing in target: 1
- Records missing in source: 1
- Records with field discrepancies: 1

Report saved to: reconciliation_report.csv

Candidates should handle potential errors gracefully, such as file not found, invalid CSV format, etc., and provide meaningful error messages to guide users.