Practice Python Exam: Banking Program

This assignment is entirely optional and designed to give you practice writing code and applying lessons and topics for the Python exam.

The Assignment

This Python exam will involve implementing a *bank program that manages bank accounts* and allows for deposits, withdrawals, and purchases.

The program will initially load a list of accounts from a .txt file, and deposits and withdrawals from additional .csv files. Then it will parse and combine all of the data and store it in a dictionary.

Steps

1. Complete the required functions
    a. Implement all of the functions defined in the *bank_accounts.py*
        i. Docstrings have already been provided
        ii. You can create any number of helper functions (with docstrings)
        iii. The *main* function has already been implemented for you
        iv. Add *brief* comments to all non-trivial lines of code
    b. Complete the unit testing in *bank_accounts_tests.py*
    c. Run the full suite of unit tests in *bank_accounts_tests_full.py* to evaluate your program
2. Make sure your program and the testing file run without errors!

Evaluation

1. Does your program work as expected?
    a. Can you make a deposit to or withdrawal from one of the accounts in the .txt file? Can you make multiple purchases?
    b. Does your program print clear and useful error messages when applicable? For example, if you try to withdraw from a non-existent account.
2. Did you implement the functions correctly?
    a. Does your program successfully load and parse the 3 files: accounts, deposits, and withdrawals, and store all the data in a dictionary database?
    b. Are you reusing functions in multiple places in your program? For example, the "purchase" function can use the "withdraw" function.

3. Did you write good unit tests?
    a. Did you write at least 2 additional test cases for each function?
    b. Do you test both typical examples and edge cases?
    c. Does your program pass all of your tests?
4. Coding Style
    a. Appropriate naming of variables
    b. Naming of helper functions (with docstrings)
    c. Clear comments in your code