



COIS 2240 – Winter 2024

Assignment 3

Deadline: 9<sup>th</sup> of April by 11:59PM

Note: The deadline has been pushed back when compared to the Syllabus.

**Objective:** The objective of this assignment is to improve your knowledge of utilizing the Java Swing GUI widget toolkit.

**Program Description:** Develop a desktop application using Java Swing that simulates basic banking operations. The application will read from and write to a CSV file, simulating a database for simplicity.

**Requirements:**

**Account Creation:** Allow users to create a new bank account by providing a name, initial deposit amount, and account type (e.g., Savings, Checking). Each account should be assigned a unique account number. Write this information to the accounts.csv file, and transactions.csv.

**Deposit:** Enable users to deposit money into their accounts by entering their account number and the amount to deposit. You are not allowed to deposit more than one million dollars at a time. This action will update the Balance field in the CSV accounts.csv, and will be an entry into the transactions.csv file.

**Withdrawal:** Allow users to withdraw money from their accounts, ensuring the account balance does not go negative. This action will update the Balance field in the CSV accounts.csv, and will be an entry into the transactions.csv file.

**Balance Inquiry:** Enable users to check their account balance by entering their account number. If the account number does not exist a popup window should warn the user, and the user will be prompted to enter the account number again. To perform a balance enquiry, you must read from the accounts.csv file. A Balance enquiry is NOT a transaction for the purpose of this assignment.

**Transaction History:** Display a simple transaction history for an account. This will return all the transactions for a specific account.

**Sample accounts.csv**

Account Number,Account Holder Name,Account Type,Balance

12345,John Doe,Savings,1000

67890,Jane Smith,Checking,500

### **Sample transactions.csv**

Account Number, Action, Date

123456,Deposit 500,15-03-2024

123456,Withdraw 1000,16-03-2024

### **Error Handling:**

Implement error and exception handling for file I/O operations, invalid user inputs (e.g., non-existent account numbers, insufficient balance), and invalid data in the CSV file.

### **Testing & Documentation:**

Include comments in your code explaining your logic.

Test all normal use cases, and all the edge cases you can think of (e.g., trying to access a bank record that does not exist).

Screenshot testing is required. Your testing should follow the following format: Test number, description of the test, expected output, actual output (screenshot).

### **Professional Look & Feel:**

In this assignment, you will be graded on the amount of effort that you put into making the GUI look and feel professional. Think of the design elements, color scheme, and performance of the application when designing it.

### **Note:**

1. You are encouraged to start by sketching the GUI layout, identifying the Java Swing components needed, and then progressively implementing the features while testing frequently.
2. This assignment is independent work, each application should represent your own design and implementation.
3. Do not hardcode a file path to the CSVs, when interacting with the CSVs we are always assuming the files are in the same directory as the code file, therefore you will only need to provide the filename, and not a path.

### **Submission Instructions:**

- 1- Each class should be a separate code file.
- 2- Submit the code file for each class, in addition to 1 PDF containing all your screenshot testing.

- 3- You MUST submit your accounts.csv and transactions.csv
- 4- Please DO NOT ZIP your submission, submit each file individually in ONE submission on Blackboard by the deadline (9<sup>th</sup> of April by 11:59PM)