C201 Team Project 3

1 Goal
The purpose of this assignment is to get you familiar with class/object and vector. Your goal is to extend the bank database application you implemented in Project 1 and Project 2.

2 Details
Your final program should include five files: bank.h and bank.cpp which were implement before, project3.cpp which is based on project2.cpp, transaction.h and transaction.cpp which are new files. The details are described below.

- Do not make changes to bank.h and bank.cpp that you are going to reuse from Project 2.
- Two new files, transaction.h and transaction.cpp, are available online. transaction.h contains the declarations of three classes, bank_account, transaction, and account_analysis. transaction.cpp contains the implementation of the methods of three classes declared in transaction.h.
- Do not make changes to transaction.h. Instead, complete the method definitions in transaction.cpp.
- Start with your finished project2.cpp and create project3.cpp and add three more options to your main program:
  - Update customer information, which allows the user to change the first name and the last name of an account (the account number and balance should remain unchanged).
  - List an account information, which allows the user to list a particular account information and its transaction history.
  - Analyze the transaction history of an account, which allows the user to show the following information about an account: number of transactions, average deposit, average withdraw, maximum deposit, and maximum withdraw.
- If you did not finish Project 2, send me an email and I will give you my copy of the completed Project 2.
- You are free to choose your new team members and you must state clearly your team members at the beginning of the program.

3 Regulations
The purpose of this project is to use class/object and vector. To add the three options to the main program, you must follow the regulations described below. Otherwise you will significantly lose credit points.

- To update customer information, you need (1) retrieve the customer information (account number, first name, last name, and balance); (2) create an instance (object) of bank_account using the constructor bank_account(int, string, string, double); (3) update the object by calling methods reset_first_name(string) and reset_last_name(string); (4) write the updates back to the database by calling methods get_id(), get_first_name(), get_last_name(), and get_balance().
- To list an particular account information, you need (1) retrieve the customer information (account number, first name, last name, and balance); (2) create an instance (object) of bank_account using the constructor bank_account(int, string, string, double); (3) display the customer information by calling methods get_first_name(), get_last_name(), and get_balance(); (4) retrieve transactions (account number, transaction type, amount, date and time) about this account and for every transaction, first, create an instance (object) of transaction using the constructor transaction(int, string, double, string), and second, display the transaction by calling method output().
- To analyze the transaction history of an account, you need (1) create an object (say AA) of account_analysis using the constructor account_analysis(int); (2) retrieve transactions about this account and for every transaction (account number, transaction type, amount, date and time), first, create an object (say T) of transaction using the constructor transaction(int, string, double, string), and second, insert this object T into the vector container (all_trans) variable of AA by calling method insert_transaction(transaction); (3) display the analysis result by calling method number_transactions(), average_deposit(), average_withdraw(), max_deposit(), and max_withdraw().
4 Notes
• Some parts of the assignment are unspecified. Use your best judgment. State your assumptions clearly. Document your code properly. Check and handle all error conditions.
• You do not need to rewrite your Project 1 and Project 2 using classes. Classes are only required to be used for the new options added in Project 3.
• Anything that is not clear, please ask the instructor.

5 Submission
Upload your source code files (.cpp and .h) using canvas. One team just needs to make one submission.