Assignment 4

Exceptions

Goals:

- Write classes to generate exceptions when detecting errors.
- Write a program to handle exceptions.

Program:

Develop a program that tests an object of the Bank that has a list of BankAccount objects. The program should support the following operations:

- **Create a Bank object.**
- **Provide a menu for the user to:**
  - Add a new BankAccount to the Bank.
  - Get the balance of a BankAccount based on the account number.
  - Deposit money to an existing BankAccount.
  - Withdraw money from an existing BankAccount.
  - Find a BankAccount with the highest balance.
  - Find a BankAccount with the lowest balance.

About your implementation:

- Upload and uncompress the Assignment4.zip file from Brightspace.
- Assignment4.zip file contains BankAccount.java, Bank.java, BankAccountException.java and BankException.java.
- Complete the implementation of BankAccount class as follow:
  - Modify the BankAccount constructors to throw a BankAccountException with appropriate message, if the account number passed to the constructor is not a 4-digit number.
  - Modify the BankAccount constructor to throw a BankAccountException with appropriate message, if the initialBalance is less than zero.
  - Modify the withdraw method to throw a BankAccountException with appropriate message, if the balance becomes less than zero.

- Complete the Bank class as follow:
  - Add the deposit and withdraw method to the Bank class. These methods deposits and withdraws money from a specific BankAccount.
  - Modify the find and getMaximum methods to throw a BankException with appropriate messages, instead of returning null values.
  - Implement getMinimum and throw a BankException with appropriate message, if no BankAccount, is added to the Bank yet.

- Create a program the BankTester class with a main method, which creates a Bank object and manipulates BankAccount objects based on user input.
    - Modularize the main program.
    - In this BankTester program, you are to ask the user for input using the Scanner class.
    - Print a 'menu' for the user to know what the possible commands are.
    - Handle all exceptions generated by the BankAccount and Bank methods for invalid input.
    - In this program, most of the errors are detected by the BankAccount and Bank methods.

Submit a single ZIP (Compressed) file to D2L containing:

The source code for all three classes: BankAccount.java Bank.java and an application program like BankTester.java.

Marking Scheme

[15] BankAccount implementation

- [10] Constructors to generate Exceptions if the account number is not a 4 digit number and/or if balance is less than 0.
- [5] Modify the withdraw to generate exception if balance becomes less than 0

[30] Bank Class implementation

- [5] Generate exceptions in the find and getMaximum methods.
- [25] Implements Method
    - [10] Deposit
    - [10] Withdraw
    - [5] getMinimum

[40] CollegeTester

- [10] functionality
    - Adding a new BankAccount to the Bank.
    - Deposit, withdraw and get the balance of a BankAccount.
    - Finding a BankAccount with the highest and lowest Balance.
- [20] Handle Exceptions
    - Your program should handle all exceptions generated by the Bank and BankAccount classes graciously, display appropriate messages to the user, and continue the program.

[10] Documentation: Javadoc for all the methods headers, purpose of the program, inline comments

[5] Programming Style: Meaningful variable names and constants following the conventions, consistent indentation.