**COLLEGE OF BUSINESS EDUCATION**



**DODOMA COMPAS**

**DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY**

***LECTURER NAME*: ATUPELE CAIRO**

***STUDENT NAME:* CHIBU NKINDA**

***REGISTRATION NO:* 02.1242.01.02.2023**

***COURSE NAME:*  BIT2**

***SUBMIT DATE:* 28 JANUARY, 2025**

***NATURE OF WORK:* INDIVIDUAL ASSIGNMENT**

***NAME OF PROJECT*: TRANSACTION PROGRAM**

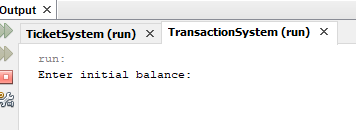
**REPORT ABOUT TRANSACTION SYSTEM**

1. **Introduction**

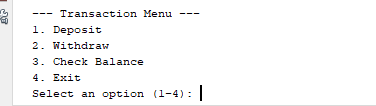
**The transaction system:**  is the system written in java programming language that help the user in money transaction. This system allow user to deposit, check balance and withdraw. In order to use Transaction system the following are the function that suppose to do:-

**2. Body explanation**.

This system perform the following function:-

**2.1 Balance Initialization**: The user starts with an initial balance. The system need the user to enter the initial balance that is greater than 0 so that it can avoid the negative balance. In this program if user enter the initial value the second option the system display the option to choose.

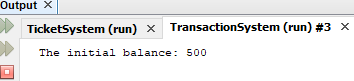
**2.2 Menu Options**: After entering the initial value then press Enter button, Graphical User Interface display the selection options by which the user wants to perform at a time. To select either 1 to Deposit money, 2 to Withdraw money, 3 to Check their balance or 4 to Exit the program.



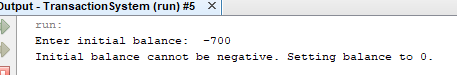
* + 1. **Deposit.** The deposit option allows the user to add money to their account. This increases the user's account balance.

**Steps:**

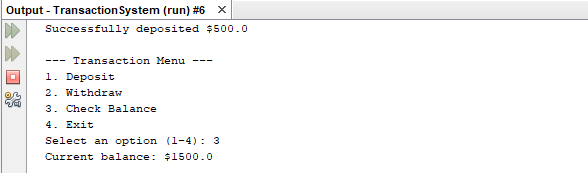
* **Input Amount:** The user specifies the amount they wish to deposit. Example: 500.



* **Validation:** Ensure the amount entered is a positive number, Optionally, verify the deposit does not exceed account or program limits.



* **Update Balance:** Add the deposit amount to the current account balance. Example: If the current balance is $1000 and the deposit is $500, the new balance becomes $1500.



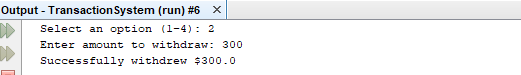
* **Confirmation:** Display a message indicating the deposit was successful. Example: "You have successfully deposited 500. Your new balance is 1500."



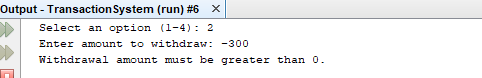
**2.2.2 Withdraw.** The withdrawal option allows the user to remove money from their account or wallet. This decreases the user's account balance.

**Steps:**

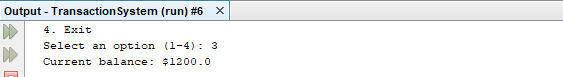
* **Input Amount:** The user specifies the amount they want to withdraw. Example: 300. The message displayed is “successfully to withdraw $300.



* **Validation:** Check if the withdrawal amount is a positive number. Ensure the user has sufficient balance for the withdrawal. Optionally, enforce limits on withdrawals per transaction, day, or account type. And if the amount is negative the program may display message “withdrawal amount must be greater than 0.”



* **Update Balance:** Deduct the withdrawal amount from the current balance. Example: If the current balance is 1500 and the withdrawal is 300, the current balance becomes 1200.

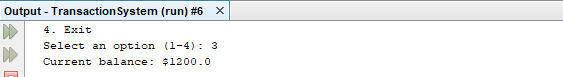


* **Confirmation:** Display a success or error message. Success Example: "You have successfully withdrawn 300. Your new balance is 1200." Also Error may occur Example: "Insufficient balance. Your current balance is 1500."

**2.2.3. Check Balance.** The check balance option allows the user to view their current account balance. This helps the user monitor their financial standing.

**Steps:**

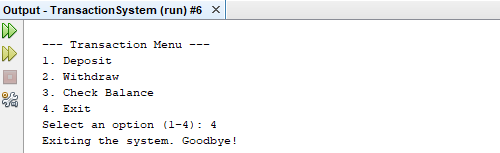
* **Retrieve Balance:** The program fetches the current balance from the database or internal records. After fetch the program display the current balance available in the account.
* **Display Balance:** Show the user their balance in a clear and concise format. Example: "Your current balance is 1200."



**Note:** A customer wants to verify the amount of money available in their account before making a transaction.

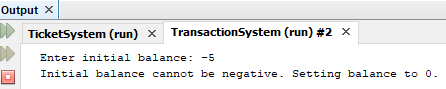
**2.2.4. Exit.**The exit option allows the user to end their session and close the program or transaction interface.

**Steps:**

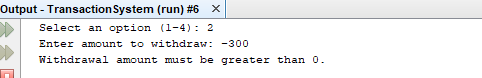
* **Terminate Session:** Safely end the user's session and close the program. Optionally, save logs of the session for security or analytics.
* **Goodbye Message (Optional):** Display a final message. Example: "Thank you for using our services. Goodbye!"
  + 

At this step a customer finishes their transactions and wants to leave the interface.

**3. Validation**: The program validates inputs to ensure that deposits and withdrawals are positive and that withdrawals do not exceed the balance. These program does not allow the negative value to be inputs of the program as shown below.



This is for withdraw amount:



**Loop**: The program runs in a loop until the user chooses to exit.

**Generally,** The program use java programming language only, is simple and clear for the user and allow continuous repetition without to leaf in the program