**Simple Bank Account System**

This Python project simulates a simple bank account system. It provides functionalities for creating accounts, depositing and withdrawing money, checking account balances, viewing transaction history, and visualizing transactions with graphs.

**Features**

**1. Account Creation**

* Users can create an account using:
  + **Account Name**
  + **Account Number** (must be alphanumeric, between 6 and 10 characters, and case-sensitive).
  + **Email ID** (optional).
* Validates inputs and ensures the account number is unique.
* Stores account information in a **dictionary** (with account number as the key) and writes it to a CSV file upon validation.
* Provides confirmation of successful account creation.

**2. Login**

* After creating an account, users can log in using their registered username and account number.
* The program reads the CSV file into a dictionary for validating login credentials.
* Upon successful login, users are redirected to their **Account Dashboard**.

**3. Account Dashboard**

* Displays current account balance.
* Allows account holders to:
  + Perform transactions (Deposit, Withdraw).
  + View transaction history.
  + Visualize their transaction graph, which shows the balance after each transaction.

**4. Deposit**

* Users can deposit amounts with the following conditions:
  + Only valid numbers are accepted.
  + Maximum amount allowed: 999,999 (6 digits).
* After a successful deposit:
  + Updates the current balance.
  + Records the transaction in the history.

**5. Withdraw**

* Users can withdraw amounts under these conditions:
  + Withdrawal amount must be less than or equal to the current balance.
* If sufficient balance is available:
  + Updates the current balance.
  + Records the transaction in the history.
* Displays an error message if the balance is insufficient.

**6. Transaction History**

* Users can view all transactions (Deposits and Withdrawals) performed after logging in.

**7. View Transaction Graph**

* Users can visualize their transaction history with a graph showing balance updates after each transaction.

**8. Log Out**

* Ends the session and redirects users to the login screen.

**Requirements**

* **Python 3.x**
* **Tkinter** (built-in with Python)
* **Matplotlib** (Install via: pip install matplotlib)
* **Pillow (PIL)** (Install via: pip install pillow)

**How to Run the Application**

Install Dependencies:

Make sure you have Matplotlib and Pilow installed

Run the Code

**Screenshots:**



