

Advanced Banking Functions:

- **Transaction History:**
 - Design a `Transaction` class to store individual transactions (date, amount, type).
 - Implement a `TransactionList` class (linked list or alternative) to manage transaction history for each account.
 - Allow users to view transaction history for a specific account within a given date range.
 - Consider filtering transactions by type (deposit, withdrawal, etc.).
- **Account Management:**
 - Design an `Account` class to represent different account types (checking, savings, loan, etc.).
 - Implement basic account functionalities like deposit, withdraw, balance inquiry, and account closure.
 - Enforce account-specific rules (e.g., minimum balance for checking, interest rates for savings).
- **User Authentication:**
 - Design a `User` class to store user information (username, password, etc.).
 - Implement secure login functionality with password hashing and salting.
 - Implement user roles and permissions (e.g., customer, admin) with access control.

Operator Overloading:

- Overload operators (e.g., `+`, `-`, `<<`, `>>`) for relevant classes like `Account` and `Transaction`.
- Allow operations like:
 - Adding/subtracting money from accounts.
 - Comparing accounts by balance or creation date.
 - Printing user-friendly representations of accounts and transactions.