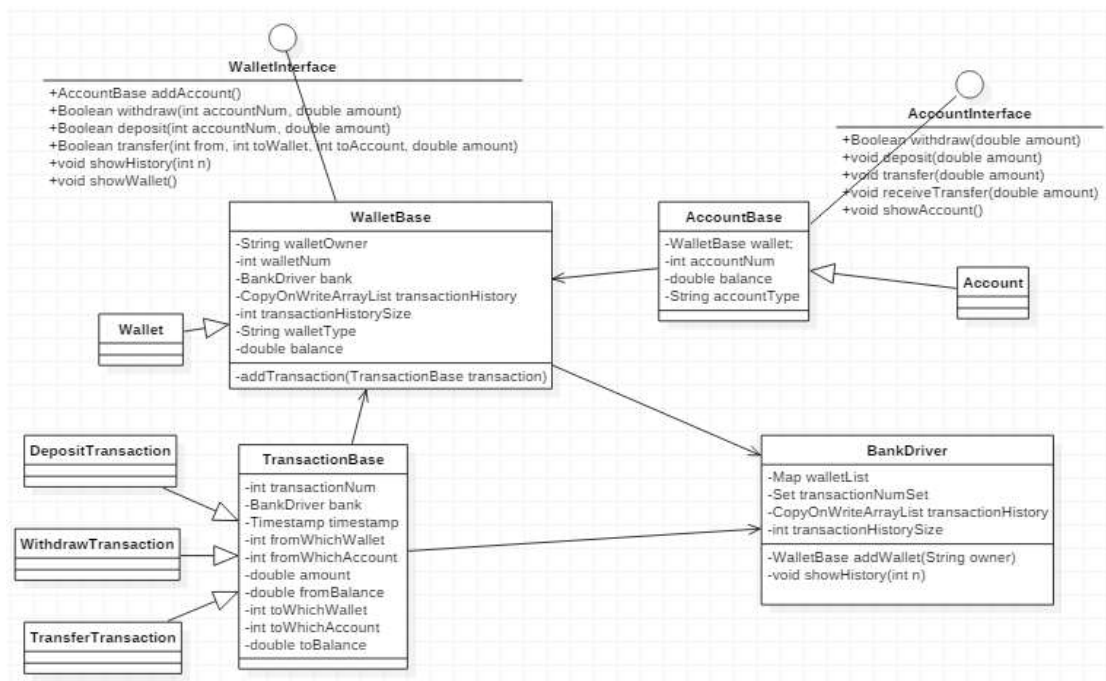


# VirtualWallet Documentation

## 1. Features

- Create wallet for a user.
- Create one or more account in a wallet
- Deposit (0-\$5000) or withdraw (0-\$800) to any account in any wallet
- Transfer (0-\$2000) between two accounts (in or not in the same wallet).
- Track balance for any account and wallet.
- Track transactions history for any account and wallet.
- Multi-threaded safe.

## 2. Structure



## 3. API

### BankDriver.class

- `WalletBase addWallet(String owner)`  
Create a Wallet for a user, assign with a random wallet number. The initial balance is 0.  
Parameter:  
    owner: the name of owner of this wallet;  
return:  
    an instance of `WalletBase`
- `void showHistory(int n)`

Show the n most recent transaction history for all wallets in this bank

Parameter:

n: the number of history to show

return:

void

### **WalletInterface.class**

- **AccountBase addAccount()**  
Create an account for a wallet, assign with a random account number. The initial balance is 0.  
Parameter:  
none  
return:  
an instance of AccountBase
- **Boolean withdraw(int accountNum, double amount)**  
Withdraw money from an account.  
If successful, add a transaction to transaction history.  
Throw InvalidWithdrawAmountException if the amount is less than 0 or bigger 800.  
Throw NoAccountNumException if the account doesn't exist.  
Throw NoEnoughBalanceException if the account doesn't have enough balance.  
Parameter:  
accountNum: account to withdraw  
amount: amount to withdraw  
return:  
true if withdraw successful  
false if failed
- **Boolean deposit(int accountNum, double amount)**  
deposit money from to an account.  
If successful, add a transaction to transaction history.  
Throw InvalidDepositAmountException if the amount is less than 0 or bigger 5000.  
Throw NoAccountNumException if the account doesn't exist.  
Parameter:  
accountNum: account to deposit  
amount: amount to deposit  
return:  
true if deposit successful  
false if failed
- **Boolean transfer(int from, int toWallet, int toAccount, double amount)**  
Transfer money between account.  
If successful, add a transaction to transaction history.  
Throw InvalidDepositAmountException if the amount is less than 0 or bigger 2000.

Throw `NoAccountNumException` if one of the accounts doesn't exist.

Throw `NoEnoughBalanceException` if the transfered account doesn't have enough balance.

Parameter:

from: transfered account

toWallet: recipient wallet

toAccount; recipient account

amount: amount to transfer

return:

true if transfer successful

false if failed

- `void showHistory(int n)`

Show the n most recent transaction history for all account in this wallet

Parameter:

n: the number of history to show

return:

void

- `void showWallet()`

Show all accounts in this wallet.

Parameter:

None.

return:

void

### **AccountInterface.class**

- `Boolean withdraw(double amount)`

Withdraw money from this account.

Throw `NoEnoughBalanceException` if the account doesn't have enough balance.

Parameter:

amount: amount to withdraw

return:

true if withdraw successful

false if failed

- `void deposit(double amount)`

deposit money from to this account.

Parameter:

amount: amount to deposit

return:

void

- void transfer(double amount)  
Withdraw money from this account for transfer.  
Throw NoEnoughBalanceException if the transfered account doesn't have enough balance.  
Parameter:  
    amount: amount to transfer  
return:  
    void
- void receiveTransfer (double amount)  
same to void deposit(double amount)
- void showAccount()  
Show account information  
Parameter:  
    None.  
return:  
    void

#### 4. Exception

- a) ExceptionBase: super class for all exceptions.
- b) InvalidAmountException: super class for all invalid amount exceptions.
- c) InvalidDepositAmountException: Thrown when the deposit amount is less than 0 or bigger than 5000
- d) InvalidTransferAmountException: Thrown when the transfer amount is less than 0 or bigger than 2000
- e) InvalidWithdrawAmountException: Thrown when the withdraw amount is less than 0 or bigger than 800
- f) NoAccountNumException: Thrown when the account user tried to deposit, withdraw or transfer doesn't exist.
- g) NoWalletNumException: Thrown when the wallet user tried to deposit, withdraw or transfer doesn't exist.
- h) NoEnoughBalanceException: Thrown when the account user tried to withdraw or transfer doesn't have enough balance.

#### 5. JUnit Test

- a) BankDriverTest: Test the overall high level functionality for the whole library
- b) WalletInterfaceTest: Test for WalletInterface: deposit, withdraw, transfer, multiple time of transfer and the transaction history.
- c) AccountInterfaceTest: Test for AccountInterface: deposit, withdraw, transfer and exception throwing.
- d) ConcurrentTest: Test for muti-threaded: concurrent deposit, withdraw, transfer.