

Class Demonstration of Bank Java Project (CS591 Java OOD)

- **Class AccountType**
 - **Attribute**
 - **Checking**: enum type, it means the account is checking account
 - **Saving**: enum type, it means the account is saving account
 - **Security**: enum type, it means the account is security account
- **Class TransactionType**
 - **Attribute**
 - **SEND** : enum type, it means the transaction is to send money
 - **Receive** : enum type, it means the transaction is to receive money
 - **SELL** : enum type, it means the transaction is to sell bonds or stocks
 - **BUY** : enum type, it means the transaction is to buy bonds or stocks
 - **WITHDRAW** : enum type, it means the transaction is to withdraw money from account
 - **LOAN** : enum type, it means the transaction is to loan money from the bank
- **Class Transaction**
 - **Attribute**
 - TransactionType **type** : indicate the type of the transaction
 - String **sender** : uid of the account which sends, loans, withdraws money or buy or sell investment
 - String **receiver** : uid of the account which receives money
 - String **money** : amount of money involved in the transaction
 - String **investmentId** : sid of investment
 - double **amount** : amount of investment
 - String **date** : the time of the transaction
- **Class Investment**
 - **Attribute**
 - private String **id** : unique for each investment product
 - **Method**
 - String **getId()** : get id of the investment

- **Class Stock(extends from Investment)**
 - **Attribute**
 - double **price** : current price for stock per share
 - String **name** : name of the stock
 - **Method**
 - void **change**() : change the price of the stock as the program runs
- **Class Bond(extends from Investment)**
 - **Attribute**
 - double **weeklyInterestRate** : weekly interest rate of bond
 - double **monthlyInterestRate** : monthly interest rate of bond
 - double **yearlyInterestRate** : yearly interest rate of bond
- **Class Property**
 - **Attribute**
 - Investment **investment** : which stock / bond
 - double **amount** : stock : number of share, bond : money
 - double **interest** : bond interest rate
 - **Method**
 - boolean **add**(double am) : only can be triggered if the property is stock, add am share
 - double **sell**() : sell all bonds and return profits
 - double **sell**(double am, double price) : sell **am** share stock at **price**, and return profits
- **Class Account**
 - **Attribute**
 - double **Aid** : unique for each account
 - ArrayList<Transaction> **transactions** : record all transactions in the account
 - double **money** : balance in the account
 - double **interestRate** : interest rate of the account
 - AccountType **type** : type of account
 - **Method**
 - boolean **withdraw**(double m) : withdraw **m** money from the account, if the balance of the account is less than money **m**, return false otherwise true and write corresponding transaction record
 - boolean **loan**(double m) : loan **m** money from the account and write corresponding transaction record
 - boolean **send**(double m, String receiver) : send money **m** to account **receiver**, if the balance of the account is less than money **m**, return false otherwise true and write corresponding transaction record
 - boolean **receive**(double m, String sender) : receiver money **m** from account **sender** and write corresponding transaction record
- **Class CheckingAccount (extends from Class Account)**

- **Class SavingAccount (extends from Class Account)**
- **Class SecurityAccount (extends from Class Account)**
 - **Attribute**
 - HashMap<String, Property> **property** : <key : unique id of investment, value : Property>, record all investments owned by the account
 - **Method**
 - boolean **buyBond**(double Money, Investment i, Property.InterestRate irate, double interest) : use **Money** money buy **i** bond at **interest** interest rate, **irate** indicates whether it is weekly, monthly or yearly
 - boolean **buyStock**(Investment i, int amount, double price) : buy **amount** number of **i** stock at **price**
 - boolean **sellBond**(String sid) : sell unique id of bonds, which equals to sid, if the account does not have, return false otherwise true
 - boolean **sellStock**(String sid, double amount, double price) : sell **amount** number of stock whose id is **sid** at **price**
- **Class User**
 - **Attribute**
 - String **Uid** : unique id for a User
 - String **password** : password to the account
 - String **name** : owner name
 - HashMap<String, Account> **accounts** : <key : Aid of the account, value : account >accounts own by the user
 - static int **count** : used to generate Aid of the account, Aid = Uid + "-" + count
 - **Method**
 - ArrayList<String> **getAccountId**() : get a list of **Aid** of accounts owned by the user
 - boolean **loanMoney**(String sender, double m) : **sender** loans **m** money
 - boolean **sendMoney**(double m, String receiver, String sender) : **sender** account sends **m** money to **receive** account
 - boolean **receiveMoney**(double m, String receiver, String sender) : **receiver** account receives **m** money from **sender** account
 - boolean **withdrawMoney**(double m, String sender) : **sender** account withdraws m money from account
 - boolean **buyStock**(Investment i, int amount, double price, String sender) : **sender** account buys **amount** stock **i** at **price**

- boolean **buyBond**(double Money, Investment i, Property.InterestRate irate, double interest, String sender) : **sender** account utilizes **Money** and buys bond **i**, with interest rate **interest**
- boolean **sellBond**(String sender, String sid) : account **sender** sell bond whose id equals to **sid**
- boolean **sellStock**(String sender, String sid, double amount, double price) : account **sender** sells **amount** share of stock whose id equals to **sid** at **price**
- boolean **closeAccount**(String sender) : close Account **sender**

- Class Bank

- Attribute

- HashMap<String, User> **users** : <key : uid, value : user>, record all information of users
- HashMap<String, Investment> **investments** : <key : sid, value : stock / bond>, record all investment products provided by the bank
- static int **userCount** : used to generate user id
- static int **investmentCount** : used to generate investment id

- Method

- boolean **addStock**(double p, String n) : add stock whose name is **n** and price is **p**
- boolean **addbond**() : add a new bond
- String **createAccounts**(String name, String password) : create an user account whose name is **name** and password is **password** , return **Uid** of the account
- boolean **login**(String Uid, String password) : log in account whose uid is **Uid** with **password**
- boolean **close**(String Uid) : close account whose uid is **Uid**
- boolean **withdrawMoney**(String Aid, double money) : Withdraw **money** from account **Aid**
- boolean **loanMoney**(String Aid, double money) : Loan **money** from account **Aid**
- boolean **sendMoney**(double m, String receiver, String sender) : Account **sender** sends **m** money to Account **receiver**
- boolean **buyBond**(double Money, String sid, Property.InterestRate irate, double interest, String sender) : Account **sender** buys bond whose id is **sid** and interest rate is **interest** with money **Money**

- boolean **buyStock**(String sid, int amount, double price, String sender) :
Account **sender** buys **amount** share of stock whose id is **sid** at **price**
- boolean **sellBond**(String sender, String sid) : Account **sender** sells bond
whose id is **sid**
- boolean **sellStock**(String sender, String sid, double amount, double price) :
Account **sender** sells stock **amount** share of stock whose id is **sid** at **price**
- String **getInvestmentInfo**() : get all the info of investments provided by the
bank