

Simple Banking System

authored by Yuan, Yifan **BuptID** 2015213364 **QMID** 151006353

Intro

A simple banking system is to be developed with the intention of providing a generic, reusable system from which to develop more realistic systems.

Bank can use this tool to have interaction with funds management.

Javaw.exe Interface in Eclipse

```
[INFO] -----  
[INFO] Banking System  
[MENU] 1. user application  
[MENU] 2. open the account  
[MENU] 3. close the account  
[MENU] 4. deposit  
[MENU] 5. withdraw  
[MENU] 6. suspend  
[MENU] 7. release  
[MENU] 8. mannual update  
[INFO] -----  
Input:|
```

Features of Banking System

the simple interface present the main functions of Banking System:

- User Application
- Open types of accounts
- Close the accounts
- Deposit and withdraw
- Suspend and Release the users' power
- Update the accounts automatically
- Using .XML files

What are listed above will be described in details.

User application

People will want to have accounts in our bank. First thing they need to do is application.

People need to filling the blanks in format, such as

```
1 [INFO] Format: name address age
2 Input: yifan bupt 18 // U input parameters here.
```

Thus u will see (if u have good credit)

```
1 [INFO] Name:yifan Address:bupt Age:18
2 [INFO] system has create a new user 0005
3 // the banking system has built the logfiles of yours
```

Open types of accounts

Next, we can continue to enter the account what u want to open. And the computer will ask some information. U can use the id generated in last step . (e.g 0005)

```
1 [INFO] Format: your userid, the type
2 [INFO] 1. current 2. junior 3. saver
3 Input:0005 1
```

```
1 [INFO] accNo: 0005c balance: 0.0 overdraftlimit: 500.0
2 [INFO] set your private PIN
3 Input:
```

After the first step input the type of account, the system will want to learn the PIN u need to set.

Here I will set PIN to 1998.

```
[MENU] 5. withdraw
[MENU] 6. suspend
[MENU] 7. release
[MENU] 8. mannual update
[INFO] -----
Input:2
[INFO] Format: your userid, the type
[INFO] 1. current 2. junior 3. saver
Input:0005 1
[INFO] accNo: 0005c balance: 0.0 overdraftlimit: 500.0
[INFO] set your private PIN
Input:1998
[INFO] PIN has been recorded
```

Close the accounts

Individual can enter the account id generated in last step and PIN he/she set in the past to close the accounts.

```

[MENU] 1. user application
[MENU] 2. open the account
[MENU] 3. close the account
[MENU] 4. deposit
[MENU] 5. withdraw
[MENU] 6. suspend
[MENU] 7. release
[MENU] 8. mannual update
[INFO] -----
Input:3
[INFO] Format: your accId, PIN
Input:0005c 1998
[INFO] accNo: 0005c has been deleted.

```

PIN is the key to author the system to operate, and it's important.

Deposit and withdraw

Deposit money in bank is a normal behavior.

```

1 [INFO] Format: your accId, PIN
2 Input: 0005c 1998

```

System only need to confirm the PIN and account Id to be authored.

Additionally, bank will ask some specific information to locate the files.

```

1 [INFO] Format: the amount, type, date(if saverAccount)
2 [INFO] 1. cash 2. cheque
3 Input:10000 1 // I use the way of cash

```

Pay attention I use the Current Account, So no date parameters are needed.

```

[MENU] 5. withdraw
[MENU] 6. suspend
[MENU] 7. release
[MENU] 8. mannual update
[INFO] -----
Input:4
[INFO]
Format: your accId, PIN
Input:0005c 1998
[INFO] Format: the amount, type, date(if saverAccount)
[INFO] 1. case 2. cheque
Input:10000 1
[INFO] accNo: 0005c balance: 10000.0 overdraftlimit: 500.0

```

Suspend and Release the users' power

System may need to suspend some accounts using this feature.

```
1 [INFO] Format: userId
2 Input: 0005
```

In this mode, Bank can use its administrative power so it doesn't need to use PIN.

somebody may noticed that I input `0005`, but not the `0005c`. personally speaking, bank need to understand that in normal life suspending an user is more reasonable.

```
[MENU] 1. user application
[MENU] 2. open the account
[MENU] 3. close the account
[MENU] 4. deposit
[MENU] 5. withdraw
[MENU] 6. suspend
[MENU] 7. release
[MENU] 8. mannual update
[INFO] -----
Input:6
[INFO] Format: userId
Input:0005
[INFO] user power suspended
```

Update the accounts automatically

System can use a timer to check the checks, the save or overdraft.

the System provide two ways to update them.

- automatically
- manually

I put these code blocks in Bank.java.

```
1 /**
2  * an automatic bot can check the accounts using an extra thread
3  * @see #cheque_update()
4  * @see #overdraft_update()
5  * @see #save_update()
6  */
7 public void autobot (){
8     Calendar calendar = Calendar.getInstance();
9     calendar.set(Calendar.HOUR_OF_DAY, 0); // hour
10    calendar.set(Calendar.MINUTE, 0);    // minute
11    calendar.set(Calendar.SECOND, 0);    // control second
12
13    Date time = calendar.getTime();
14
15    Timer timer = new Timer();
16    timer.scheduleAtFixedRate(new TimerTask(){
17        public void run(){
18            save_update();
19            cheque_update();
```

```

20         overdraft_update();
21     }
22     },time,1000*3600*24);
23 }

```

the parameter `1000*3600*24` means that the system will check 1 time each day.

System also can use menu 8 `[MENU] 8. mannual update` to update manually.

Using .XML files

My system use the XML to store the information. They are readable and easily to organize in coding.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
- <bank>
  - <user id="0001" name="yifan" credit="true" age="19"
    address="bupt">
    - <currentAccount state="true" pin="0000" accNo="0001c">
      <balance>12897.0</balance>
      <overdraft>500.00</overdraft>
    </currentAccount>
  </user>
  - <user id="0002" name="mzy" credit="true" age="15"
    address="bupt">
    - <juniorAccount state="true" pin="1996" accNo="0002j">
      <balance>11000.0</balance>
    </juniorAccount>
    - <saverAccount state="true" pin="1996" accNo="0002s">
      <balance>0.00</balance>
    </saverAccount>
  </user>
  - <user id="0003" name="yifan" credit="true" age="19"
    address="bupt">
    - <saverAccount state="true" pin="1998" accNo="0003s">
      <balance>0.00</balance>
    </saverAccount>
  </user>
  - <user id="0004" name="0" credit="true" age="0" address="0">
    - <saverAccount state="true" pin="0" accNo="0004s">
      <balance>0.00</balance>
    </saverAccount>
  </user>
  - <user id="0005" name="yifan" credit="false" age="18"
    address="bupt">
    - <currentAccount state="true" pin="1998" accNo="0005c">
      <balance>10000.0</balance>
      <overdraft>500.00</overdraft>
    </currentAccount>
  </user>
</bank>

```

.xml files are listed in the files documentary.

Other tips

In fact, somebody may input some wrong parameters in the process of inputs.

I have finished some methods to cope with it.

```
1 public void checkpara(int k,String[] args)
2 public static void checkUserId(String userId)
3 public static void checkAccId(String accId)
4 public static void checkPin(String accId,String pin)
```

They are listed in the `Main.java`