Basics: From C to C++

Computer Programming for Engineers (DSAF003-42)
Fall, 2021

Assignment : PA2

Instructor:
Youngjoong Ko (nlp.skku.edu)

PA2

- **There is a Bank that can create up to 5 Accounts. Each account has information about user_id, name, money.**

- **Bank provide five basic functions**
  1. Create Account
  2. Send Money
  3. Deposit Money
  4. Withdraw Money
  5. Print Account Informations

- **User cannot directly access the account, and All command must be access through the bank functions.**

- **Implement bank system using given Bank class and Account class code.**

Main function

- **This program get User Commands and runs until Exit Program command. (Assumed that the user inputs only integer command.)**

- **Each command means the following actions**

  0 : Create new account

  1 : Deposit Money

  2 : Withdraw Money

  3 : Send Money

  4 : Exit Program

  Others : Wrong Input (print Error Message)

- **Only call Bank Object's Functions in main function.**
- **After a command (including "Wrong Input") executed, Print Information of all created account.**

Bank class

- **Functions**

  1. Create Account

     Create Account Object in Bank class.

     Account's Variables (User id, Name, Money) should be initialized.

     - User id increases by 1 from 0
     - Name should be initialized to user input
     - Money should be initialized to 0.

  2. Deposit Money

     Get two integer user input (user id, money).

     Increase user id's account money.

Bank class

- **Functions**

  3. Withdraw Money

     Get two integer user input (user id, money).

     Decrease user id's account money if account have enough money.

     If account doesn't have enough money, print error message.

  4. Send Money

     Get two integer user input (user id1, user id2, money).

     If user id1's account doesn't have enough money, print error message.

     If user id1's account have enough money, decrease user id1's account money and Increase user id2's account money

  5. Print Account Informations

Account class

- **Variables**

  User Id (int)

  Name (string)

  Money (int)

- **Changing or Initializing variable of Account must all be done at the Account's function.**

Must follow rules(total 5pt)

- **Only the function of Bank Object can be called in the main function.**
  - Access to account's variable in main function (-1pts)
  - Call account's function in main function (-1pts)

- **Don't access Account's variable at Bank's Function.**
  - Access to account's variable in Bank's function (-1pts)
  - Use Constructor for Initializing Account's variables
  - Use Account's function for Change Account's variables

- **Use cout format in slide 16~.**

Examples

- **Command 0**
  - No Max user

  < Bank state (before) >

User id	Name	Money
-	-	-

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 0

Enter user name : Kim

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0

0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 0

Enter user name : Lee

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 0
```

Examples

- **Command 0**
  - No Max user

  < Bank state (before) >

User id	Name	Money
0	Lee	0
1	Lee	0
2	Lee	0
3	Lee	0
4	Lee	0

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 0

Error : Max user...

 ######## Bank User Info ########
User : 0        Name : Lee      Money : 0
User : 1        Name : Lee      Money : 0
User : 2        Name : Lee      Money : 0
User : 3        Name : Lee      Money : 0
User : 4        Name : Lee      Money : 0
```

Examples

- **Command 1**
  - Normal Input

< Bank state (before) >

User id	Name	Money
0	Kim	0

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 1

Enter User idx : 0
Enter money : 1000

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 1000
```

  - Wrong User id

< Bank state (before) >

User id	Name	Money
0	Kim	1000

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 1

Enter User idx : 1
Enter money : 1000

Error : No exist user ...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 1000
```

Examples

- **Command 2**
  - Normal input

  < Bank state (before) >

User id	Name	Money
0	Kim	1000

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 2

Enter User idx : 0
Enter money : 500

Try to withdraw 500 from USER0 ...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 500
```

  - Wrong User id

  < Bank state (before) >

User id	Name	Money
0	Kim	500

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 2

Enter User idx : 1
Enter money : 500

Error : No exist user ...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 500
```

Examples

- **Command 2**
  - Not enough Money

< Bank state (before) >

User id	Name	Money
0	Kim	500

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 2

Enter User idx : 0
Enter money : 1000

Try to withdraw 1000 from USER0 ...

Error : Not enough money...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 500
```

Examples

Red box is input value

- **Command 3**
  - Normal input

  < Bank state (before) >

User id	Name	Money
0	Kim	500
1	Lee	0

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 3

Enter User idx (From): 0
Enter User idx (To): 1
Enter money : 500

Try to send 500 from USER0 to USER1 ...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 500
```

  - Wrong User id

  < Bank state (before) >

User id	Name	Money
0	Kim	0
1	Lee	500

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 3

Enter User idx (From): 1
Enter User idx (To): 3
Enter money : 500

Error : No exist user ...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 500
```

Examples

- **Command 3**
  - Not enough Money

< Bank state (before) >

User id	Name	Money
0	Kim	0
1	Lee	500

Red box is input value

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 3

Enter User idx (From): 1
Enter User idx (To): 0
Enter money : 1000

Try to send 1000 from USER1 to USER0 ...

Error : Not enough money...

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 500
```

Examples

- **Command 4**

  < Bank state (before) >

User id	Name	Money
0	Kim	0
1	Lee	0

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 4

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 0


프로세스가 종료되었습니다.
```

- **Others**

  < Bank state (before) >

User id	Name	Money
0	Kim	0
1	Lee	0

```
0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit
Enter Command : 7

Command Error!

 ######## Bank User Info ########
User : 0        Name : Kim      Money : 0
User : 1        Name : Lee      Money : 0
```

cout format

- **Use follow formats to print. (If you have to print some variables value, fill empty space)**

< Error Message >

```
cout << "Error : Not enough money..." << endl;
cout << "Error : Max user..." << endl;
cout << "Error : No exist user ..." << endl;
cout << "Command Error!" << endl;
```

cout format

< Input >

```
cout << "Enter user name : ";
cout << "Enter Command : ";
cout << "Enter User idx : ";
cout << "Enter money : ";
cout << "Enter User idx (From): ";
cout << "Enter User idx (To): ";
```

< Information >

```
cout << " ######## Bank User Info ######## " << endl;
cout << "Try to send " <<    << " from USER" <<    << " to USER" <<    << " ... " << endl;
cout << "Try to withdraw " <<    << " from USER" <<    << " ... " << endl;
cout << "User : " <<    << "\tName : " <<    << "\tMoney : " <<    << endl;
cout << "0. Create Account | 1. Deposit | 2. Withdraw | 3. Send | 4. Exit" << endl;
```