HW 4

ELEC 3150 – Object Oriented Programming (Fall 2023)

Nick Cebula

Q1: main.cpp   (Chose to use default class constructor balance=500 at start)

```cpp
#include <iostream>
#include <string>
#include "Account.h"
using std::cin;
using std::cout;
using std::string;
using std::endl;

int main() {
    // Create an Account object on the heap
    Account* ptr_Account = new Account();

    // Check balance
    cout << "Balance: $" << ptr_Account->checkbalance() << endl;

    // Deposit $1000
    ptr_Account->deposit(1000);
    // Check balance
    cout << "Balance: $" << ptr_Account->checkbalance() << endl;

    // Withdraw $300
    ptr_Account->withdraw(300);
    // Check balance
    cout << "Balance: $" << ptr_Account->checkbalance() << endl;

    // Delete the object
    delete ptr_Account;
    return 0;
}
```

Account.h:

```cpp
#pragma once
#include <iostream>
#include <string>
using std::string;
class Account
{
    //attributes
    string name;
    int balance;
public:
    //constructor
    Account();
    Account(string in_name, int in_balance);
    //method
    void deposit(int in_money);
    void withdraw(int out_money);
    int checkbalance();
    string showname();
    //destructor
    ~Account();
};
```

Account.cpp

```cpp
#include "Account.h"

//constructor
Account::Account() {
    name = "NA";
    balance = 500;
}
Account::Account(string in_name, int in_balance) {
    in_name = name;
    in_balance = balance;
}
//method
void Account::deposit(int in_money) {
    balance += in_money;
}
void Account::withdraw(int out_money) {
    balance -= out_money;
}
int Account::checkbalance() {
    return balance;
}
string Account::showname() {
    return name;
}
//destructor
Account::~Account() {

}
```

Q2:

Main.cpp:

```cpp
Bank_Admin admin;
admin.update_name(*ptr_Account,"Peter");
cout << "Updated Account name: " << ptr_Account->showname() << endl;

// Add $10 interest to the account
admin.interest(*ptr_Account, 10);
cout << "Updated Account balance after interest: $" << ptr_Account->checkbalance() << endl;

// Change total balance of the account to $50
admin.update_balance(*ptr_Account, 50);
cout << "Updated Account balance: $" << ptr_Account->checkbalance() << endl;

// Delete the object
delete ptr_Account;
return 0;
```

Bank_Admin.h:

```cpp
#pragma once
#include <iostream>
#include <string>
using std::string;
#include "Account.h"


class Bank_Admin
{
private:
    //attributes
    string name;
    int balance;
public:
    //constructors

    //methods
    void update_name(Account& user, string in_name);
    void interest(Account& user, int interest_rate);
    void update_balance(Account& user, int in_balance);

    //destructor
    ~Bank_Admin();
};
```

Bank_Admin.cpp

```cpp
#include "Bank_Admin.h"

void Bank_Admin::update_name(Account& user, string in_name){
    user.name = in_name;
}

void Bank_Admin::interest(Account& user, int interest_rate){
    user.deposit(interest_rate);
}

void Bank_Admin::update_balance(Account& user, int in_balance){
    user.balance = in_balance;
}

Bank_Admin::~Bank_Admin(){
}
```

Results (q1&2 together):

```
Balance: $500
Balance: $1500
Balance: $1200
Updated Account name: Peter
Updated Account balance after interest: $1210
Updated Account balance: $50
```