

Name - Jay Bhattarai
USN - IBM19IS198

```
#include <iostream>
#include <string>
#include <math.h>
using namespace std;
class Account
{
public:
    string customerName, typeofAccount;
    int accountNumber, balance;
    Account(string cname, string account, int
            accNumber, int bal)
    {
        customerName = cname;
        type of Account = account;
        accountNumber = accNumber;
        balance = bal;
    }
    Account(): customerName("NULL"), typeofAccount
        ("NULL"), accountNumber(0), balance(0){}

    void enterData();
    void displayBalance();
    void depositBalance();
};
```



```
void Account::enterData()
```

```
{
```

```
    cout << "\n Enter customer name" << endl;
```

```
    cin >> customerName;
```

```
    cout << "\n Enter account number" << endl;
```

```
    cin >> accountNumber;
```

```
    cout << "\n Enter account type" << endl;
```

```
    cin >> typeOfAccount;
```

```
    cout << "\n Enter balance" << endl;
```

```
    cin >> balance;
```

```
}
```

```
void Account::displayBalance()
```

```
{
```

```
    cout << "\n Customer Name" << customerName;
```

```
    cout << "\n Account Number" << accountNumber;
```

```
    cout << "\n Type" << typeOfAccount;
```

```
    cout << "\n the balance is : " << balance;
```

```
}
```

```
void Account::depositBalance()
```

```
{
```

```
    int amount;
```

```
    cout << "\n Enter amount to deposit" << endl;
```

```
    cin >> amount;
```

```
    balance = balance + amount;
```

```
}
```


class SavingsAccount : public Account.

{

int interest;

public:

SavingsAccount(string cname, string account, int accNumber,
int bal, int intr): Account(cname, account,
accNumber, bal)

{

interest = intr;

}

SavingsAccount(): interest(0){};

int compoundInterest()

{

int time, rate;

rate = 10;

cout << "\n Enter time" << endl;

cin >> time;

interest = balance * pow(1 + rate / 100.0, time) - balance;

return interest;

}

void updateBalance()

{

balance = balance + compoundInterest();

cout << "The balance is: " << balance << endl;

}

void withdrawl()

{

int amount;

cout << "\n Enter amount to withdraw" << endl;

cin >> amount;

if (balance ~~>~~ amount)

{

balance = balance - amount;

}

else

{

cout << "The amount cannot be withdrawn" << endl;

}

}

};

class Current Account: public Account.

{

int checkBook; service charge;

int minBalance = 500;

public:

CurrentAccount(string cname, string account, int
accNumber, int bal, int intr): Account(cname,
account, accNumber, bal).

{

checkbook = intr;

}

CurrentAccount(): checkBook(0) {};


```
int mainBalance()
```

```
{
```

```
    int penalty = 1;
```

```
    if (balance <= minBalance)
```

```
    {
```

```
        serviceCharge = 50;
```

```
        balance = balance - serviceCharge;
```

```
        penalty = 0;
```

```
    }
```

```
    else
```

```
    {
```

```
        cout << "\n No service charge" << endl;
```

```
    }
```

```
    return penalty;
```

```
}
```

```
void withdrawl()
```

```
{
```

```
    int amount;
```

```
    cout << "\n Enter the amount to withdraw" << endl;
```

```
    cin >> amount;
```

```
    int checkPenalty = mainBalance();
```

```
    if (checkPenalty == 1)
```

```
    {
```

```
        if (balance >= amount)
```

```
            balance = balance - amount;
```

```
    }
```



```
else  
{
```

```
cout << "\n The amount cannot be withdrawn" << endl;
```

```
}
```

```
}
```

```
};
```

```
int main()
```

```
{
```

```
SavingsAccount s1;
```

```
CurrentAccount c1;
```

```
s1.enterData();
```

```
s1.displayBalance();
```

```
s1.depositBalance();
```

```
s1.displayBalance();
```

```
s1.updateBalance();
```

```
s1.withdraw();
```

```
s1.displayBalance();
```

```
c1.enterData();
```

```
c1.displayBalance();
```

```
c1.depositBalance();
```

```
c1.displayBalance();
```

```
c1.withdraw();
```

```
c1.displayBalance();
```

```
return 0;
```

```
};
```