

Java Lab Program 5 } Current Account and Savings Account

```
import java.util.Scanner
```

```
class account
```

```
{
```

```
String name;
```

```
int accno;
```

```
String type;
```

```
double balance;
```

```
account (String name, int accno, String type, double balance)
```

```
{
```

```
    this.name = name;
```

```
    this.accno = accno;
```

```
    this.type = type;
```

```
    this.balance = balance;
```

```
}
```

```
void deposit(double amount)
```

```
{
```

```
    balance = amount;
```

```
}
```

```
void withdraw (double amount)
```

```
{
```

```
    if ((balance - amount) > 0)
```

```
    {
```

```
        balance = amount;
```

```
    }
```

```
    else
```

```
    {
```

```
        System.out.println ("insufficient balance, can't withdraw");
```

```
    }
```

```
}
```



```
void display ()
```

```
{
```

```
    System.out.println ("name: " + name + " account  
    " type: " + type + " balance: " + balance);
```

```
}
```

```
}
```

```
class savAcc extends account
```

```
{
```

```
    private static double rate = 5;
```

```
    savAcc (String name, int accno, double balance)
```

```
{
```

```
        super (name, accno, "savings", balance);
```

```
}
```

```
void interest ()
```

```
{
```

```
    balance += balance * (rate) / 100;
```

```
    System.out.println ("balance : " + balance);
```

```
}
```

```
}
```

```
class curAcc extends account
```

```
{
```

```
    private double minBal = 500;
```

```
    private double serviceCharges = 50;
```

```
    curAcc (String name, int accno, double balance)
```

```
{
```

```
        super (name, accno, "current", balance);
```

```
}
```



```
void checkmin()
```

```
{
    if (balance < minbal)
```

```
{
```

```
    System.out.println("balance is less than min  
balance, service charges imposed: " + servicecharge);
```

```
    System.out.println("balance is: " + balance);
```

```
}
```

```
}
```

```
class accountMain
```

```
{
```

```
    public static void main (String a[])
```

```
{
```

```
        Scanner s = new Scanner (System.in);
```

```
        System.out.println ("enter the name: ");
```

```
        String name = s.next();
```

```
        System.out.println ("enter the type (current/savings): account number");
```

```
        String type = s.next();
```

```
        System.out.println ("enter the account number");
```

```
        int acno = s.nextInt();
```

```
        System.out.println ("enter initial balance");
```

```
        double balance = s.nextDouble();
```

```
        int ch;
```

```
        double amount1, amount2;
```

```
        account acc = new account (name, acno, type, balance);
```

```
        savAcct sa = new savAcct (name, acno, balance);
```

```
        curAcct ca = new curAcct (name, acno, balance);
```

```
        while (true)
```

```
{
```

```
        if (acc.type.equals ("savings"))
```

```
{
```



```
System.out.println("\n Menu \n 1. deposit 2. withdraw  
3. compute interest 4. display");
```

```
System.out.println("Enter the choice");
```

```
ch = s.nextInt();
```

```
switch (ch)
```

```
{
```

```
case 1: System.out.println("Enter the amount");
```

```
amount1 = s.nextInt();
```

```
sa.deposit(amount1);
```

```
break;
```

```
case 2: System.out.println("Enter the amount");
```

```
amount2 = s.nextInt();
```

```
sa.withdraw(amount2);
```

```
break;
```

```
case 3: sa.interest();
```

```
break;
```

```
case 5: System.exit(0)
```

```
default: System.out.println("Invalid Input");
```

```
break;
```

```
}
```

```
}
```

```
else
```

```
{
```

```
System.out.println("\n Menu \n 1. deposit 2. withdraw 3. display
```

```
System.out.println("Enter the choice");
```

```
ch = s.nextInt();
```

```
switch (ch)
```

```
{
```


case 1: System.out.println("enter the amount");
amount1 = s.nextInt();
ca.deposit(amount1);
break;

case 2 System.out.println("enter the amount");
amount2 = s.nextInt();
ca.withdraw(amount2);
ca.checkMin();
break;

case 3: ca.display();
break;

case 4 system exit(0);
default System.out.println("invalid input");
break;

7

7

7

7

7



output

Enter the type of account (Savings or Current)

Savings

Enter the customer name:

Arana

Enter account number

12345

MENU

1. Deposit 2. Withdraw 3. Compute interest 4. Display
5. Exit

Enter choice:

1

Enter the deposit amount

2000

Deposit of 2000 successful

MENU

1. Deposit 2. Withdraw 3. Compute interest 4. Display Account
5. Exit details

Enter choice:

2

Enter withdrawal amount:

200

Withdrawal of 200 successful

Menu

1. deposit 2. withdraw 3. Compute interest 4. Display
5. Exit

Enter choice 3

Interest of 40 computed and added to account
Enter choice : 4

Customer Name Piviana

Account number : 12345

Account type : Savings

Balance : 1590.00

Menu

1. deposit 2. withdraw 3. Compute interest 4. Display Account
details

5. Exit

Enter choice

5

~~Exiting~~

✓

16/01/24