

Lab Program-5

```
import java.util.Scanner;  
class account.
```

```
{  
    private String name;  
    private long account_number;  
    private int account_type;  
    double balance;  
    void get_data()  
    {
```

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

```
        System.out.println("Enter your Name");
```

```
        name = ss.next();
```

```
        System.out.println("Enter the Account Number"  
                             );
```

```
        account_number = ss.nextLong();
```

```
        System.out.println("Choose the account  
                             type: 1. Savings account  
                                   2. current account");
```

```
        account_type = ss.nextInt();
```

```
    }  
    int return_account_type()
```

```
    {  
        return account_type;  
    }
```

```
}
```

```
class savings extends account
```

```
{
```

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

```
    double amount;
```



```
void get_sav_balance()
```

```
{
```

```
    System.out.println("Enter the Amount to  
    be placed in your savings account");
```

```
    amount = ss.nextDouble();
```

```
    balance += amount;
```

```
}
```

```
void display_sav_balance()
```

```
{
```

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

```
}
```

```
void compute_sav_interest()
```

```
{
```

```
    System.out.println("\n ** Calculating Compound  
    Interest **");
```

```
    System.out.print("Enter Annual Interest rate  
    : ");
```

```
    float rate = ss.nextFloat();
```

```
    System.out.print("Enter time in years: ");
```

```
    float time = ss.nextFloat();
```

```
    System.out.print("Enter principle: ");
```

```
    float principle = ss.nextFloat();
```

```
    float I = (float) (principle * (Math.pow  
        ((1 + rate / (12 * 100)), (12 * time)))  
        - principle);
```

```
    System.out.println("The Compound Interest is  
    : " + I);
```

```
}
```

```
void withdrawal_sav()
```

```
{
```

```
    System.out.println("Enter the amount to  
    be withdrawn");
```



```
amount = ss.next Double();  
balance = balance - amount;  
}
```

```
{
```

```
class current extends account
```

```
{
```

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

```
double amount;
```

```
final double min min_balance = 500;
```

```
void get_cur_balance()
```

```
{
```

```
System.out.println("Enter the amount  
to be placed in your current account");
```

```
amount = ss.next Double();
```

```
balance += amount;
```

```
}
```

```
void display_cur_balance()
```

```
{
```

```
System.out.println("Balance = " + balance);
```

```
}
```

```
void compute_cur_service_charges()
```

```
{
```

```
if (balance < min_balance)
```

```
{
```

```
System.out.println("service tax of  
rs. 100 shall be levied");
```

```
balance = balance - 100;
```

```
}
```

```
else
```

```
{
```

```
System.out.println("Minimum balance  
is maintained");
```

```
}
```

(29)


```
}  
void withdrawl - cur()
```

```
{
```

```
System.out.println("Enter the amount to  
be withdrawn");
```

```
Amount = ss.nextDouble();
```

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

```
}
```

```
}  
class bank_main
```

```
{
```

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

```
{  
Scanner ss = new Scanner (System.in);
```

```
int type;
```

```
System.out.println("Enter the bank detail");
```

```
account acc = new account();
```

```
acc.get_data();
```

```
type = acc.return_account_type();
```

```
if (type == 1)
```

```
{  
System.out.println("SAVINGS ACCOUNT");
```

```
savings sav = new savings();
```

```
sav.get_sav_balance();
```

```
sav.display_sav_balance();
```

```
System.out.println("Do you want to  
Calculate Interest or not? In If yes press 1  
else 0");
```

```
int ch = ss.nextInt();
```

```
if (ch == 1)
```

```
{
```



```
    sav.compute - sav.interact ();
```

```
}
```

```
    sav.display - sav.blnc ();
```

```
    sav.withdrawl - sav ();
```

```
    sav.display - sav.blnc ();
```

```
}
```

```
if ( type == 2)
```

```
{
```

```
    System.out.println ("CURRENT ACCOUNT");
```

```
    Current cur = new Current ();
```

```
    cur.get cur.balance ();
```

```
    cur.display - cur.blnc ();
```

```
    cur.compute - cur.service charges ();
```

```
    cur.display - cur.blnc ();
```

```
    cur.withdrawl - cur ();
```

```
    cur.display - cur.blnc ();
```

```
}
```

```
}
```

```
}
```