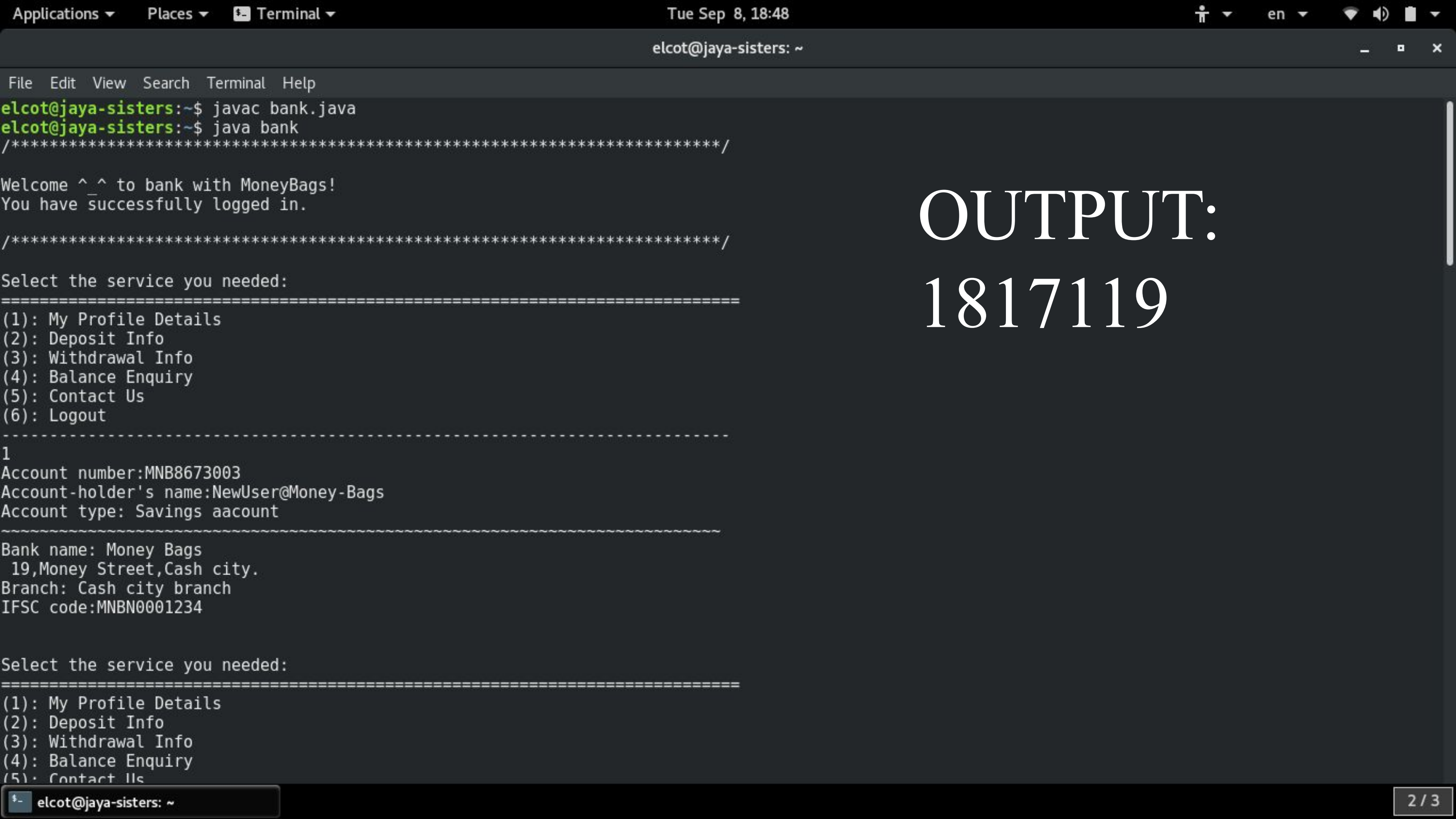
Basic Banking Application in JAVA

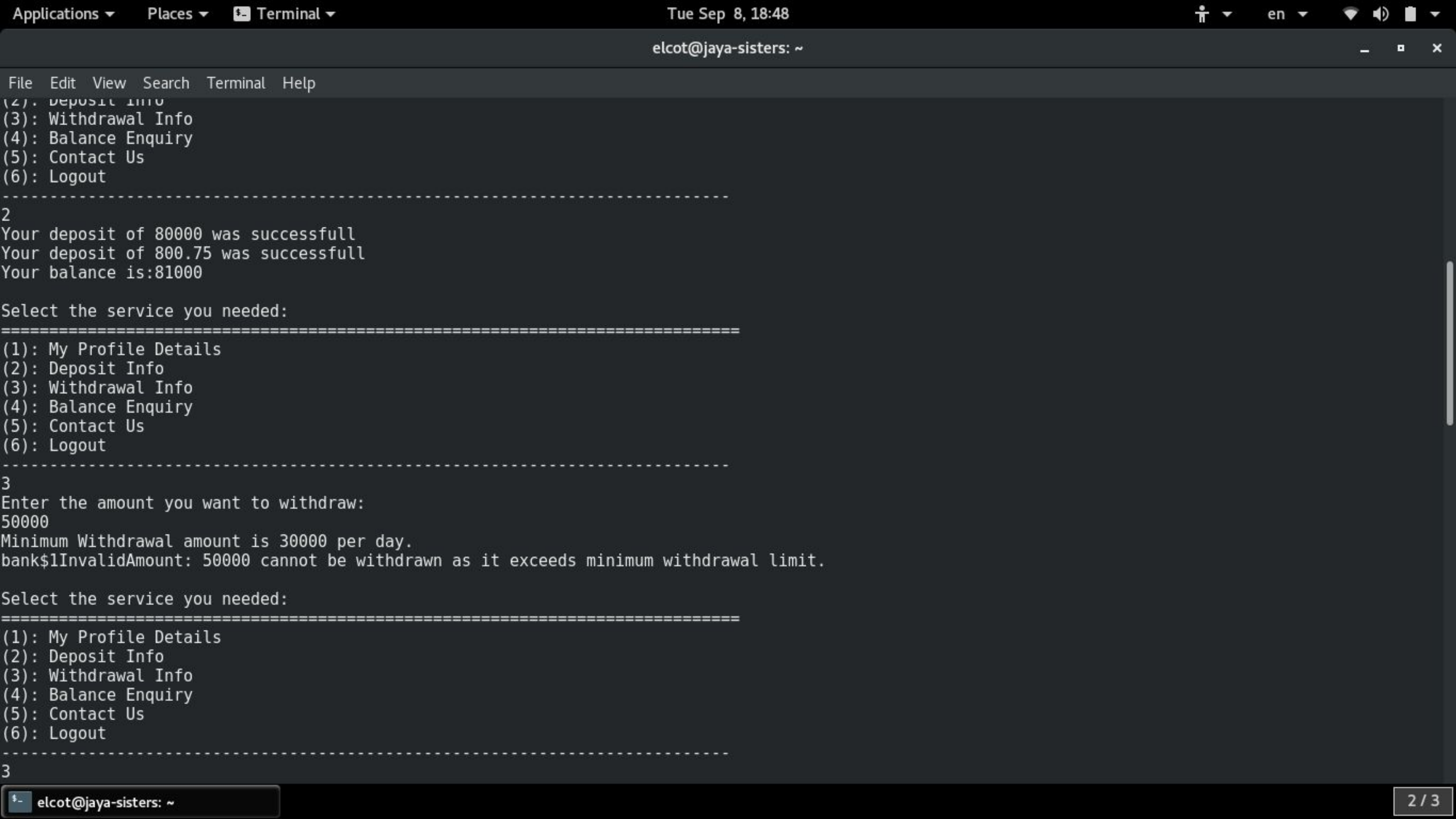
- JAYANTHI T (1817119)



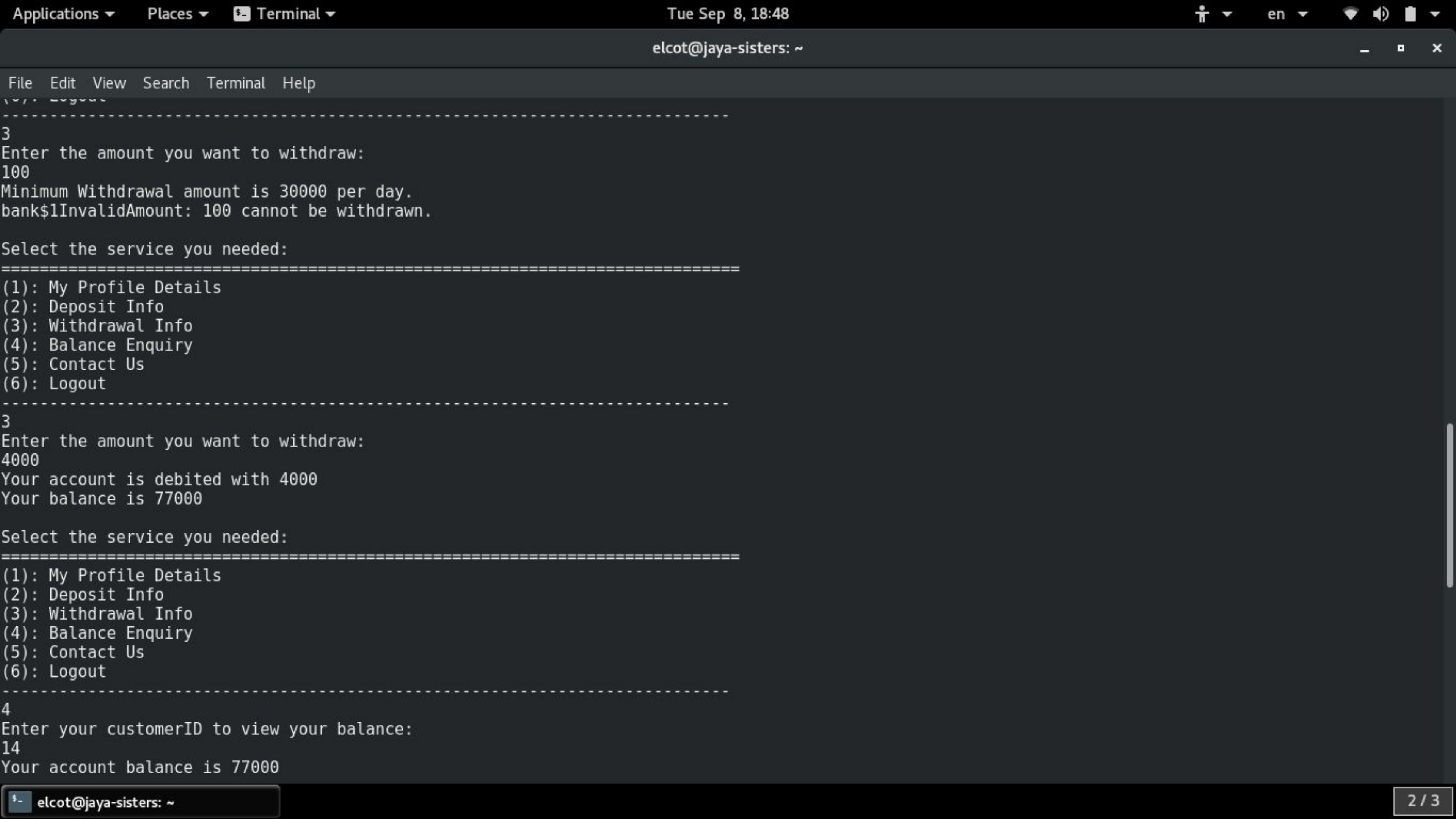
```
elcot@jaya-sisters:~$ javac bank.java
elcot@jaya-sisters:~$ java bank
/***************************************************************************/

Welcome ^_^ to bank with MoneyBags!
You have successfully logged in.

/***************************************************************************/

Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
1
Account number:MNB8673003
Account-holder's name:NewUser@Money-Bags
Account type: Savings aacount
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Bank name: Money Bags
 19,Money Street,Cash city.
Branch: Cash city branch
IFSC code:MNBN0001234


Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
```

OUTPUT:

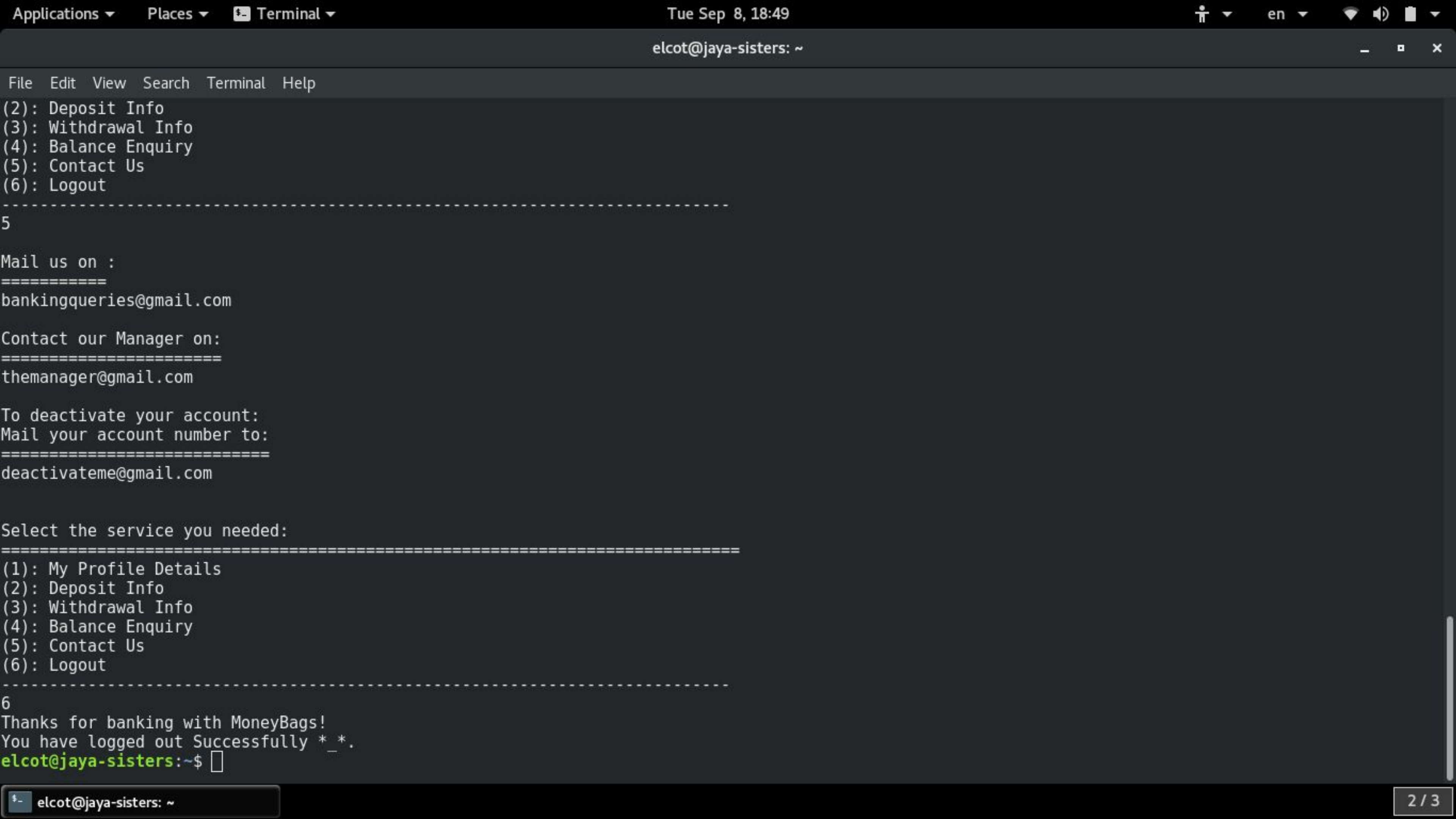
1817119



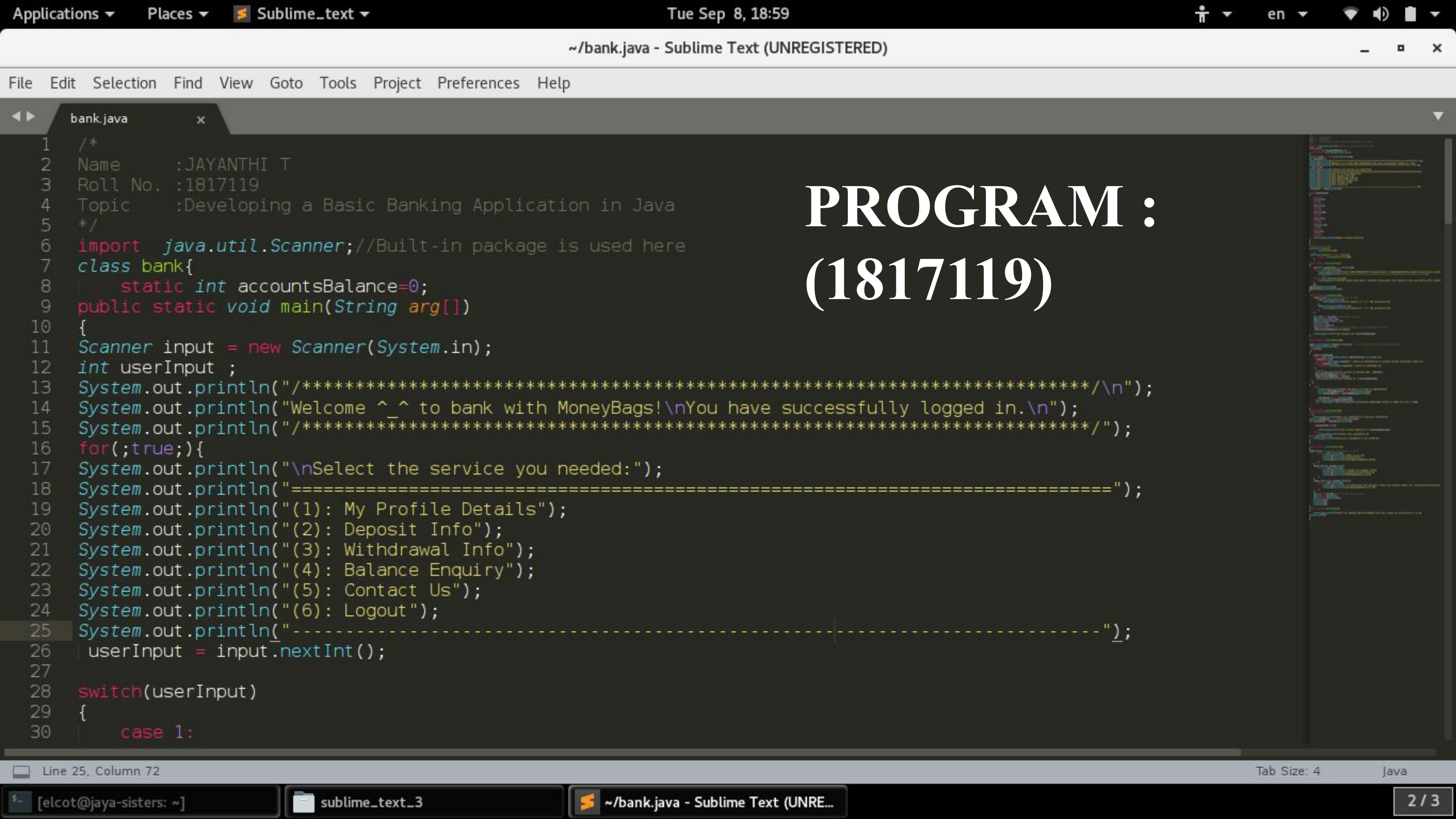
```
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
2
Your deposit of 80000 was successfull
Your deposit of 800.75 was successfull
Your balance is:81000

Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
3
Enter the amount you want to withdraw:
50000
Minimum Withdrawal amount is 30000 per day.
bank$1InvalidAmount: 50000 cannot be withdrawn as it exceeds minimum withdrawal limit.

Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
3
```



```
(6): Logout
----------------------------------------------------------------------------
3
Enter the amount you want to withdraw:
100
Minimum Withdrawal amount is 30000 per day.
bank$1InvalidAmount: 100 cannot be withdrawn.

Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
3
Enter the amount you want to withdraw:
4000
Your account is debited with 4000
Your balance is 77000

Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
4
Enter your customerID to view your balance:
14
Your account balance is 77000
```

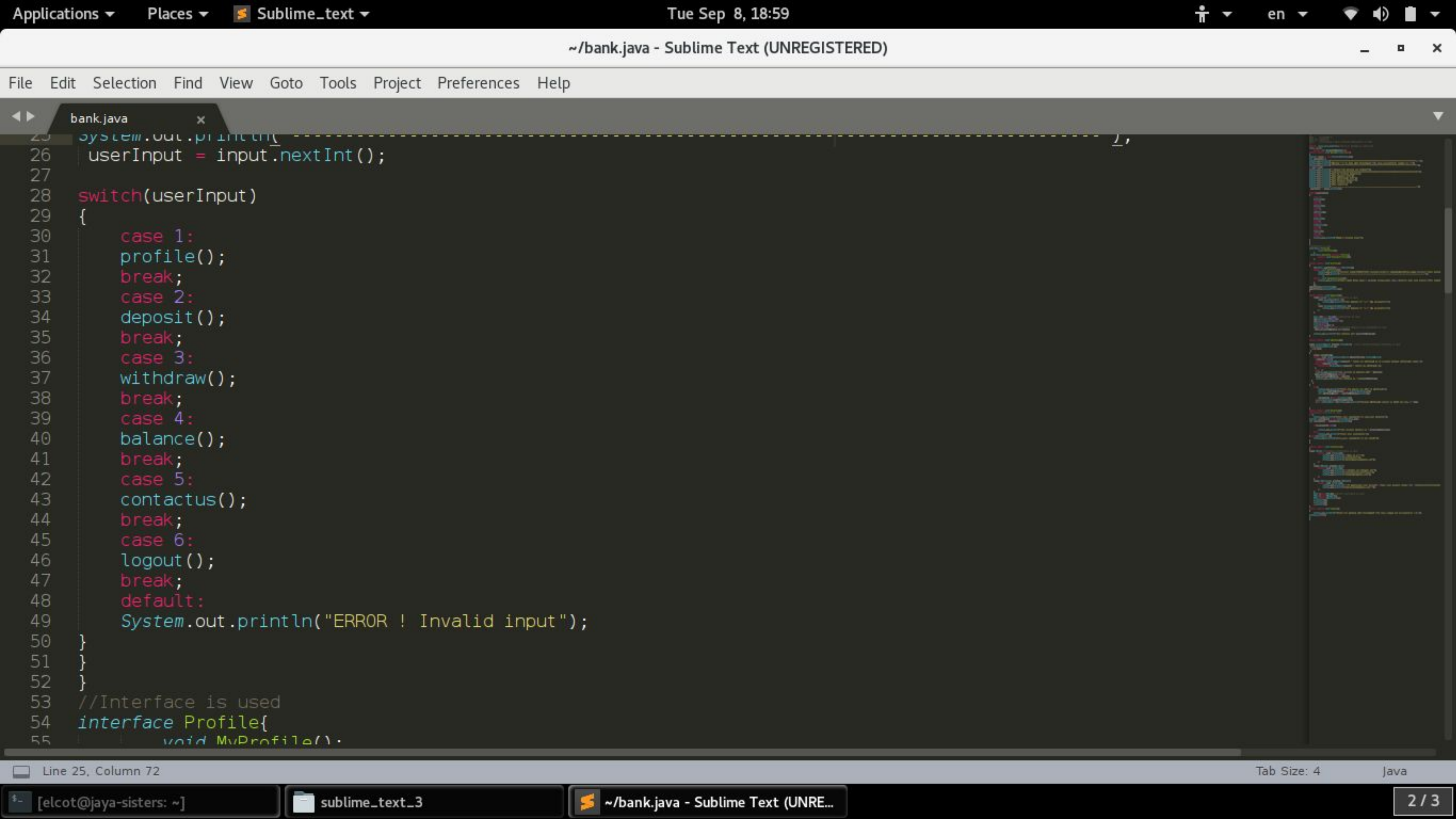


```
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
5

Mail us on :
===========
bankingqueries@gmail.com

Contact our Manager on:
=======================
themanager@gmail.com

To deactivate your account:
Mail your account number to:
============================
deactivateme@gmail.com


Select the service you needed:
=============================================================================
(1): My Profile Details
(2): Deposit Info
(3): Withdrawal Info
(4): Balance Enquiry
(5): Contact Us
(6): Logout
----------------------------------------------------------------------------
6
Thanks for banking with MoneyBags!
You have logged out Successfully *_*.
elcot@jaya-sisters:~$ 
```

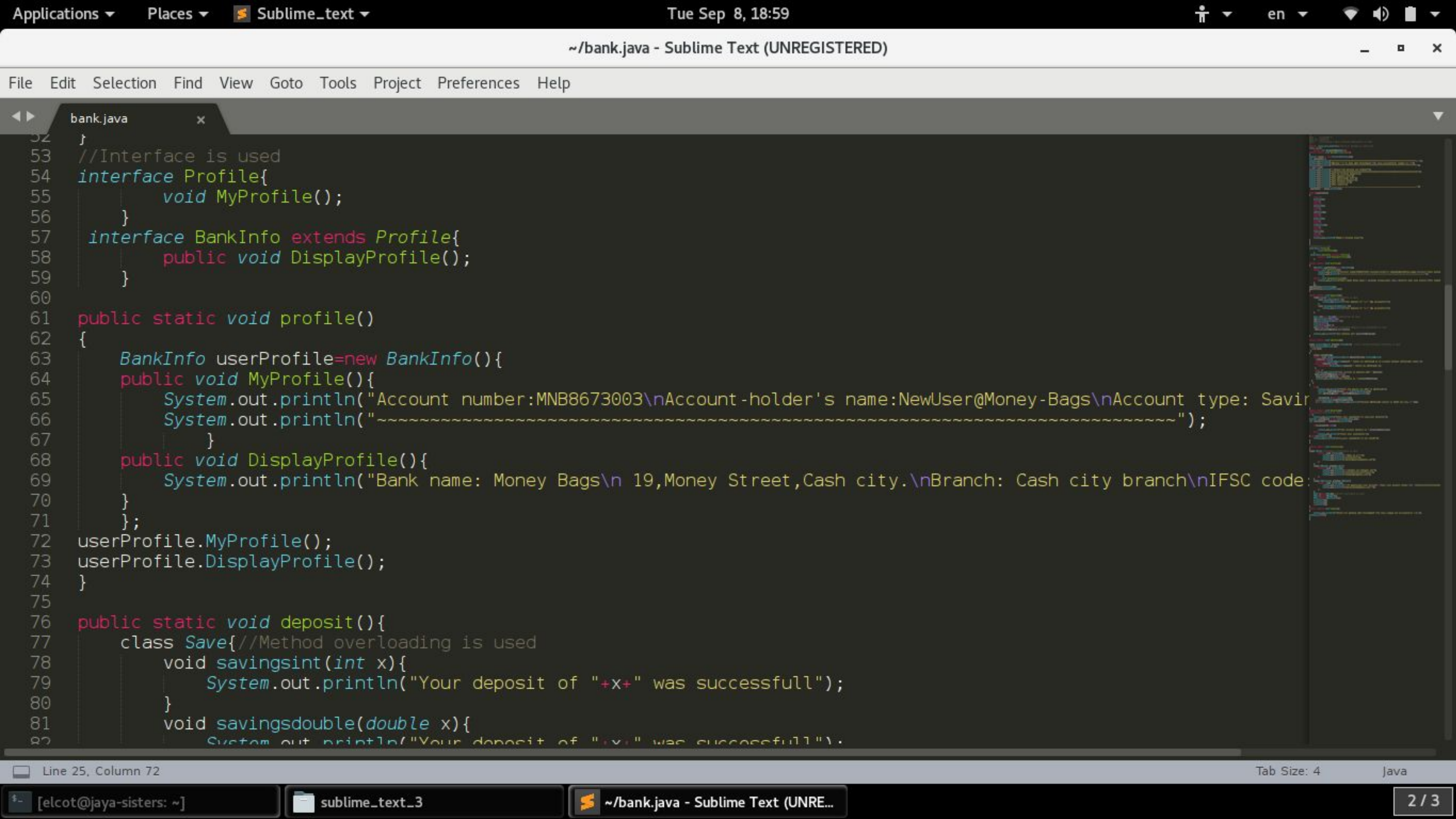


PROGRAM : (1817119)

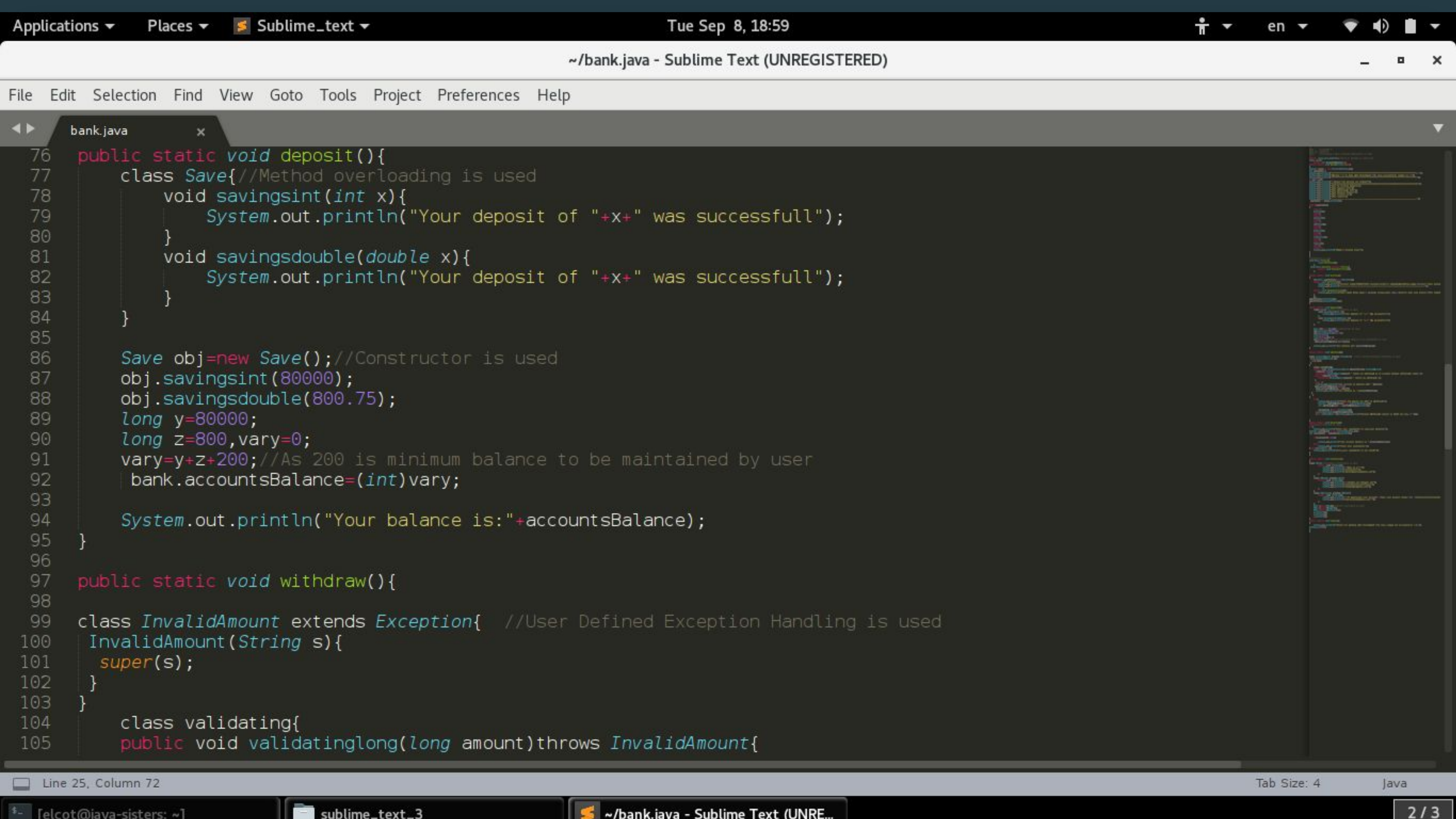
```java
/*
Name     :JAYANTHI T
Roll No. :1817119
Topic    :Developing a Basic Banking Application in Java
*/
import  java.util.Scanner;//Built-in package is used here
class bank{
    static int accountsBalance=0;
public static void main(String arg[])
{
Scanner input = new Scanner(System.in);
int userInput ;
System.out.println("/**************************************************************************/\n");
System.out.println("Welcome ^_^ to bank with MoneyBags!\nYou have successfully logged in.\n");
System.out.println("/**************************************************************************/");
for(;true;){
System.out.println("\nSelect the service you needed:");
System.out.println("=============================================================================");
System.out.println("(1): My Profile Details");
System.out.println("(2): Deposit Info");
System.out.println("(3): Withdrawal Info");
System.out.println("(4): Balance Enquiry");
System.out.println("(5): Contact Us");
System.out.println("(6): Logout");
System.out.println("----------------------------------------------------------------------------");
 userInput = input.nextInt();

switch(userInput)
{
    case 1:
```



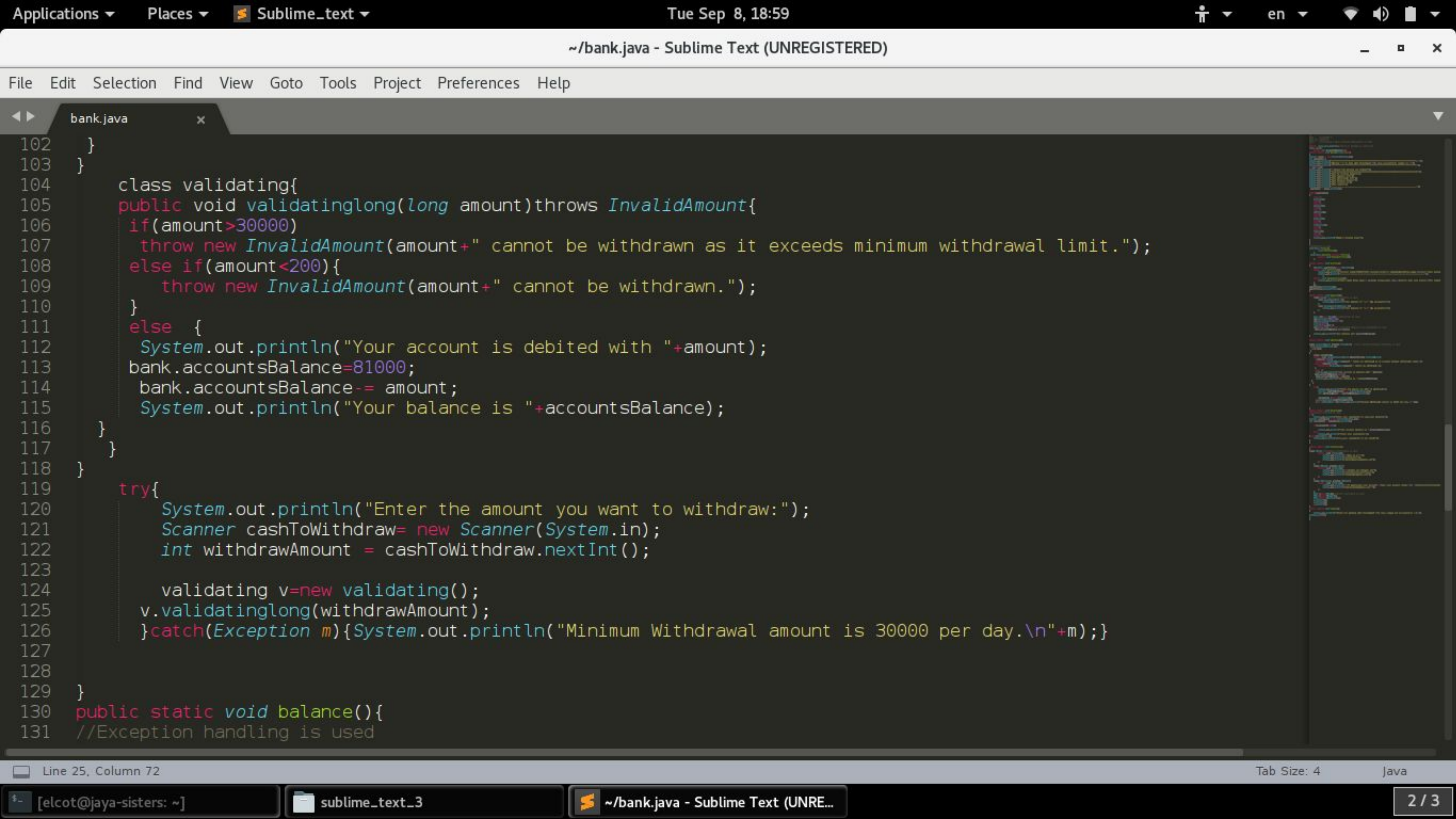
```java
System.out.println("----------------------------------------------------------------------------");
 userInput = input.nextInt();

switch(userInput)
{
    case 1:
    profile();
    break;
    case 2:
    deposit();
    break;
    case 3:
    withdraw();
    break;
    case 4:
    balance();
    break;
    case 5:
    contactus();
    break;
    case 6:
    logout();
    break;
    default:
    System.out.println("ERROR ! Invalid input");
}
}
}
//Interface is used
interface Profile{
        void MyProfile();
```



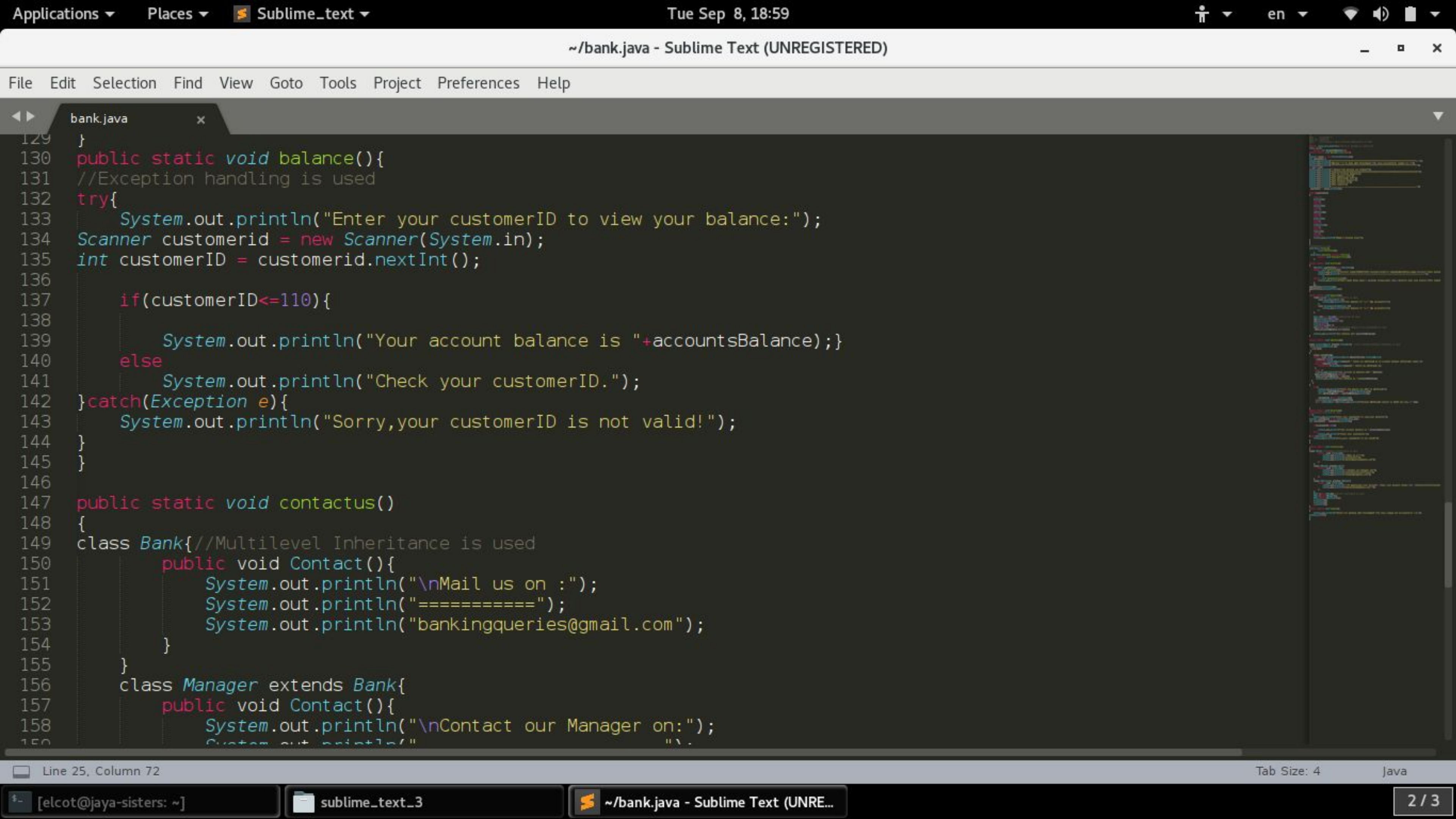
```java
}
//Interface is used
interface Profile{
        void MyProfile();
    }
 interface BankInfo extends Profile{
        public void DisplayProfile();
    }

public static void profile()
{
    BankInfo userProfile=new BankInfo(){
    public void MyProfile(){
        System.out.println("Account number:MNB8673003\nAccount-holder's name:NewUser@Money-Bags\nAccount type: Savir
        System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
            }
    public void DisplayProfile(){
        System.out.println("Bank name: Money Bags\n 19,Money Street,Cash city.\nBranch: Cash city branch\nIFSC code:
    }
    };
userProfile.MyProfile();
userProfile.DisplayProfile();
}

public static void deposit(){
    class Save{//Method overloading is used
        void savingsint(int x){
            System.out.println("Your deposit of "+x+" was successfull");
        }
        void savingsdouble(double x){
            System.out.println("Your deposit of "+x+" was successfull");
```



```java
public static void deposit(){
    class Save{//Method overloading is used
        void savingsint(int x){
            System.out.println("Your deposit of "+x+" was successfull");
        }
        void savingsdouble(double x){
            System.out.println("Your deposit of "+x+" was successfull");
        }
    }

    Save obj=new Save();//Constructor is used
    obj.savingsint(80000);
    obj.savingsdouble(800.75);
    long y=80000;
    long z=800,vary=0;
    vary=y+z+200;//As 200 is minimum balance to be maintained by user
     bank.accountsBalance=(int)vary;

    System.out.println("Your balance is:"+accountsBalance);
}

public static void withdraw(){

class InvalidAmount extends Exception{  //User Defined Exception Handling is used
 InvalidAmount(String s){
  super(s);
 }
}
    class validating{
    public void validatinglong(long amount)throws InvalidAmount{
```



```
 }
}
  class validating{
  public void validatinglong(long amount)throws InvalidAmount{
   if(amount>30000)
    throw new InvalidAmount(amount+" cannot be withdrawn as it exceeds minimum withdrawal limit.");
   else if(amount<200){
     throw new InvalidAmount(amount+" cannot be withdrawn.");
   }
   else  {
    System.out.println("Your account is debited with "+amount);
   bank.accountsBalance=81000;
    bank.accountsBalance-= amount;
    System.out.println("Your balance is "+accountsBalance);
 }
  }
}
  try{
     System.out.println("Enter the amount you want to withdraw:");
     Scanner cashToWithdraw= new Scanner(System.in);
     int withdrawAmount = cashToWithdraw.nextInt();

     validating v=new validating();
    v.validatinglong(withdrawAmount);
    }catch(Exception m){System.out.println("Minimum Withdrawal amount is 30000 per day.\n"+m);}


}
public static void balance(){
//Exception handling is used
```

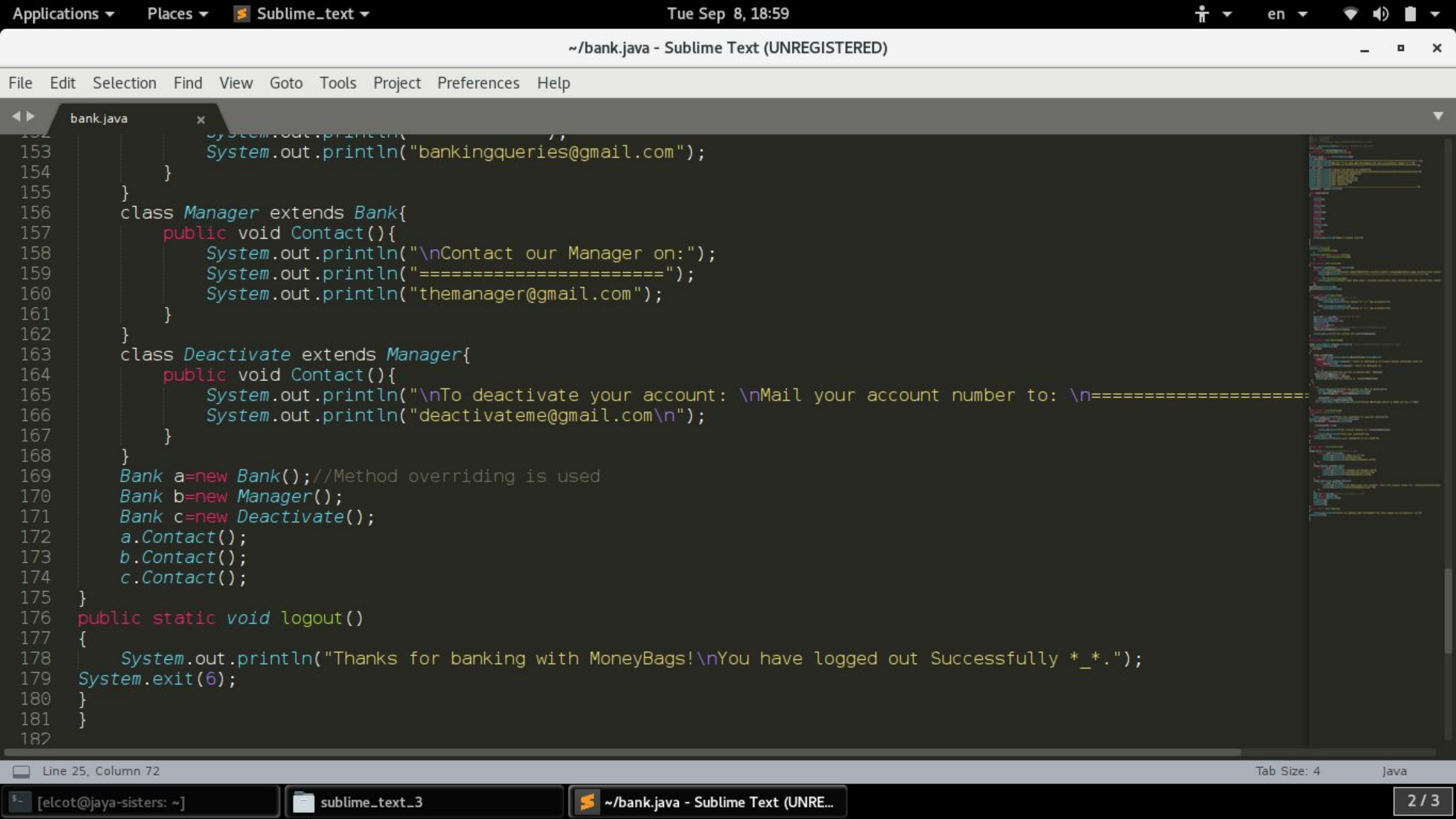


```
}
public static void balance(){
//Exception handling is used
try{
    System.out.println("Enter your customerID to view your balance:");
Scanner customerid = new Scanner(System.in);
int customerID = customerid.nextInt();

    if(customerID<=110){

        System.out.println("Your account balance is "+accountsBalance);}
    else
        System.out.println("Check your customerID.");
}catch(Exception e){
    System.out.println("Sorry,your customerID is not valid!");
}
}

public static void contactus()
{
class Bank{//Multilevel Inheritance is used
        public void Contact(){
            System.out.println("\nMail us on :");
            System.out.println("===========");
            System.out.println("bankingqueries@gmail.com");
        }
    }
    class Manager extends Bank{
        public void Contact(){
            System.out.println("\nContact our Manager on:");
```

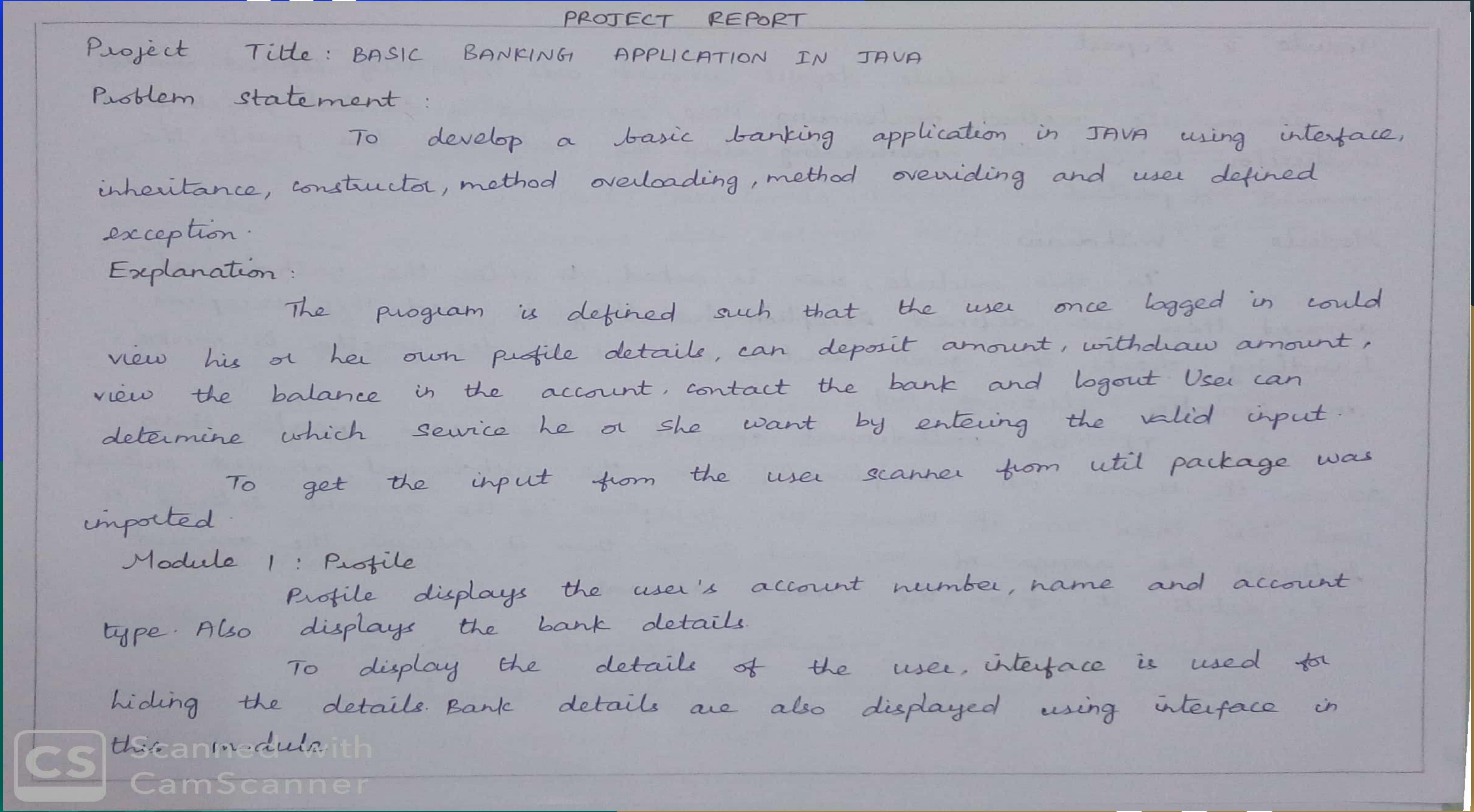
Line 25, Column 72

Tab Size: 4

Java



```
                System.out.println("bankingqueries@gmail.com");
            }
        }
        class Manager extends Bank{
            public void Contact(){
                System.out.println("\nContact our Manager on:");
                System.out.println("=======================");
                System.out.println("themanager@gmail.com");
            }
        }
        class Deactivate extends Manager{
            public void Contact(){
                System.out.println("\nTo deactivate your account: \nMail your account number to: \n====================
                System.out.println("deactivateme@gmail.com\n");
            }
        }
        Bank a=new Bank();//Method overriding is used
        Bank b=new Manager();
        Bank c=new Deactivate();
        a.Contact();
        b.Contact();
        c.Contact();
    }
    public static void logout()
    {
        System.out.println("Thanks for banking with MoneyBags!\nYou have logged out Successfully *_*.");
    System.exit(6);
    }
    }
```



PROJECT REPORT

Project Title : BASIC BANKING APPLICATION IN JAVA

Problem statement :

To develop a basic banking application in JAVA using interface, inheritance, constructor, method overloading, method overriding and user defined exception.

Explanation :

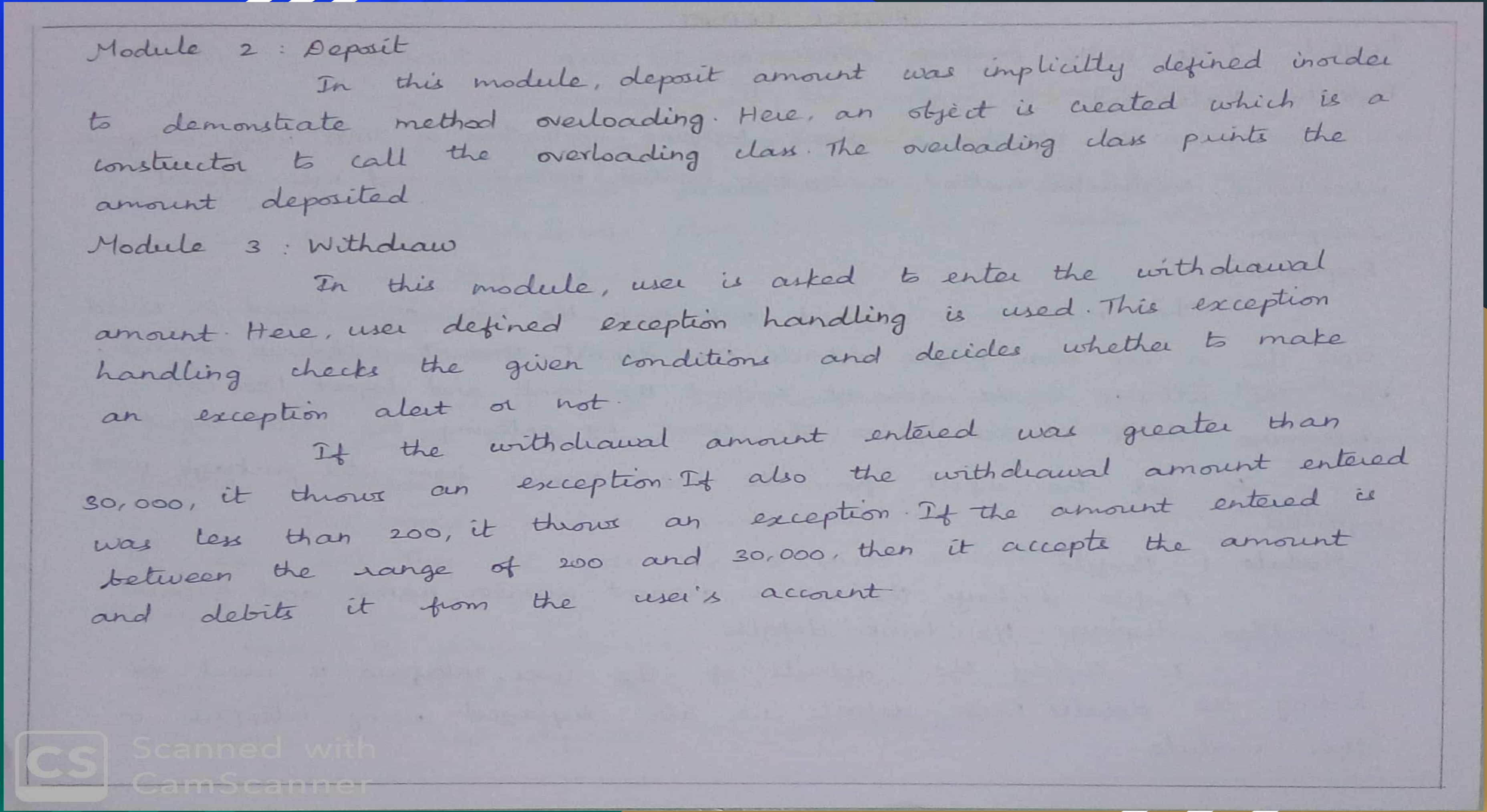
The program is defined such that the user once logged in could view his or her own profile details, can deposit amount, withdraw amount, view the balance in the account, contact the bank and logout User can determine which service he or she want by entering the valid input.

To get the input from the user scanner from util package was imported.

Module 1 : Profile

Profile displays the user's account number, name and account type. Also displays the bank details.

To display the details of the user, interface is used for hiding the details. Bank details are also displayed using interface in this module.



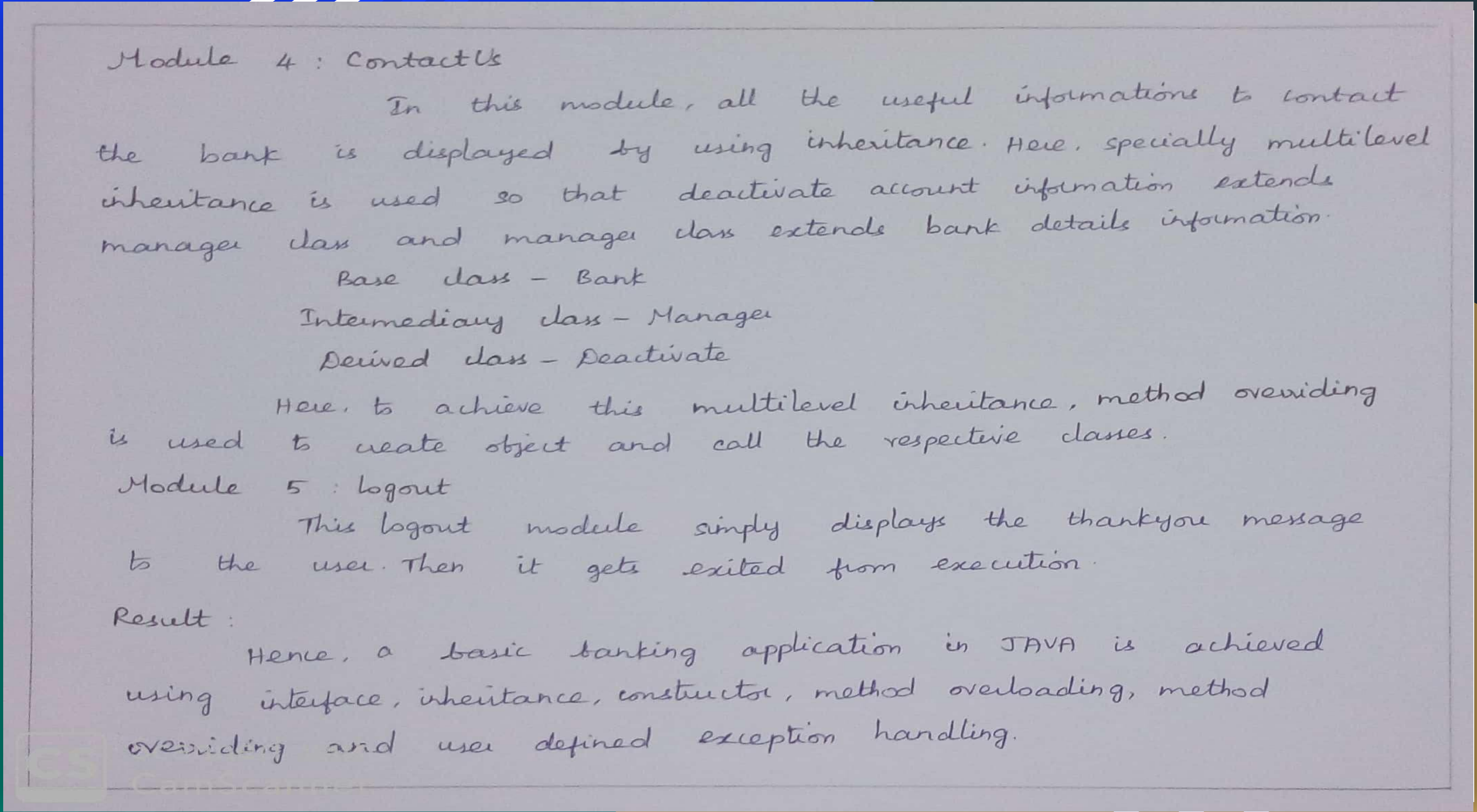
Module 2 : Deposit

In this module, deposit amount was implicilty defined inorder to demonstrate method overloading. Here, an object is created which is a constructor to call the overloading class. The overloading class prints the amount deposited

Module 3 : Withdraw

In this module, user is asked to enter the withdrawal amount. Here, user defined exception handling is used. This exception handling checks the given conditions and decides whether to make an exception alert or not.

If the withdrawal amount entered was greater than 30,000, it throws an exception. If also the withdrawal amount entered was less than 200, it throws an exception. If the amount entered is between the range of 200 and 30,000, then it accepts the amount and debits it from the user's account



Module 4 : ContactUs

In this module, all the useful informations to contact the bank is displayed by using inheritance. Here, specially multilevel inheritance is used so that deactivate account information extends manager class and manager class extends bank details information.

Base class – Bank

Intermediary class – Manager

Derived class – Deactivate

Here, to achieve this multilevel inheritance, method overriding is used to create object and call the respective classes.

Module 5 : logout

This logout module simply displays the thankyou message to the user. Then it gets exited from execution.

Result :

Hence, a basic banking application in JAVA is achieved using interface, inheritance, constructor, method overloading, method overriding and user defined exception handling.