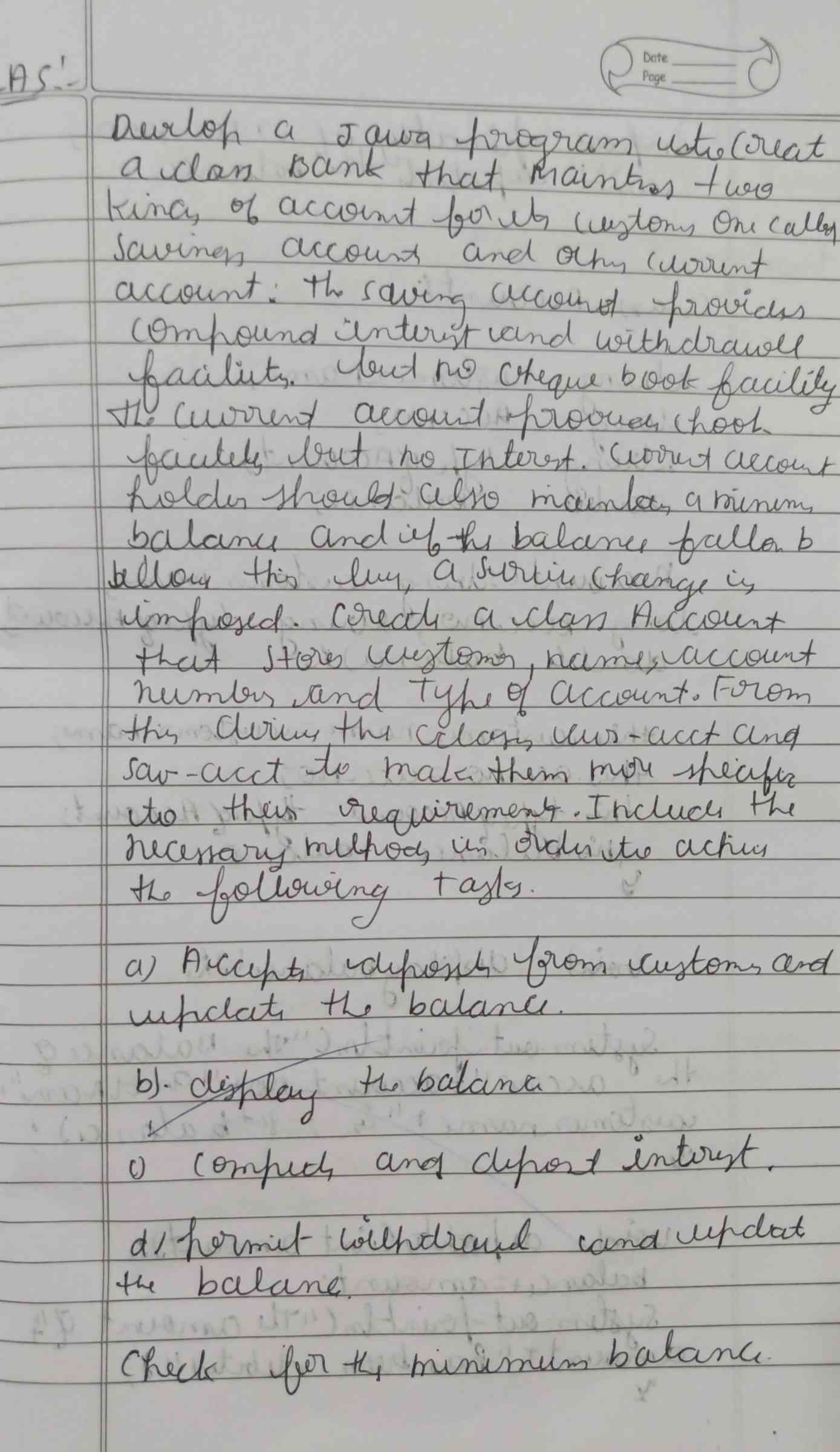
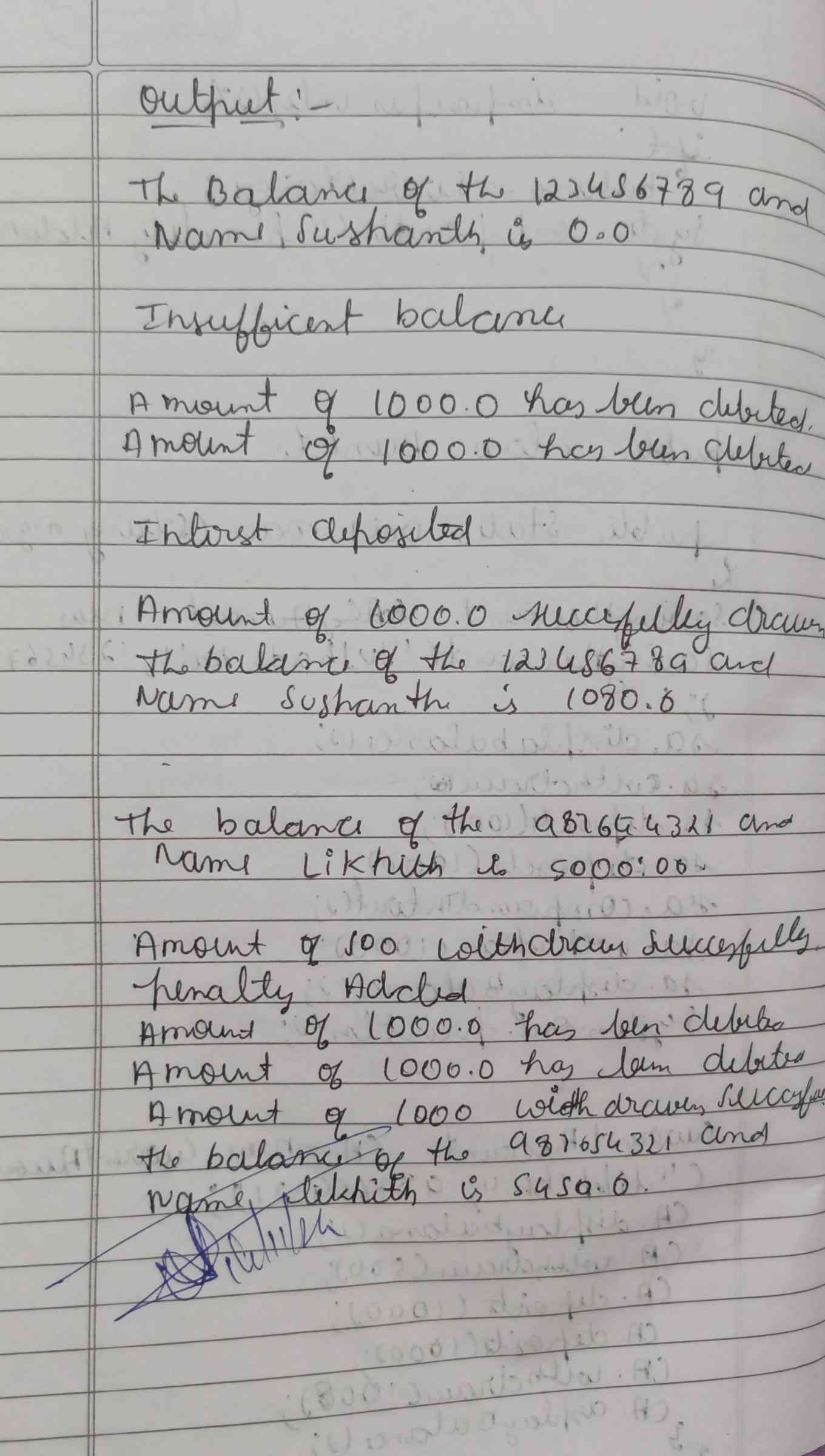
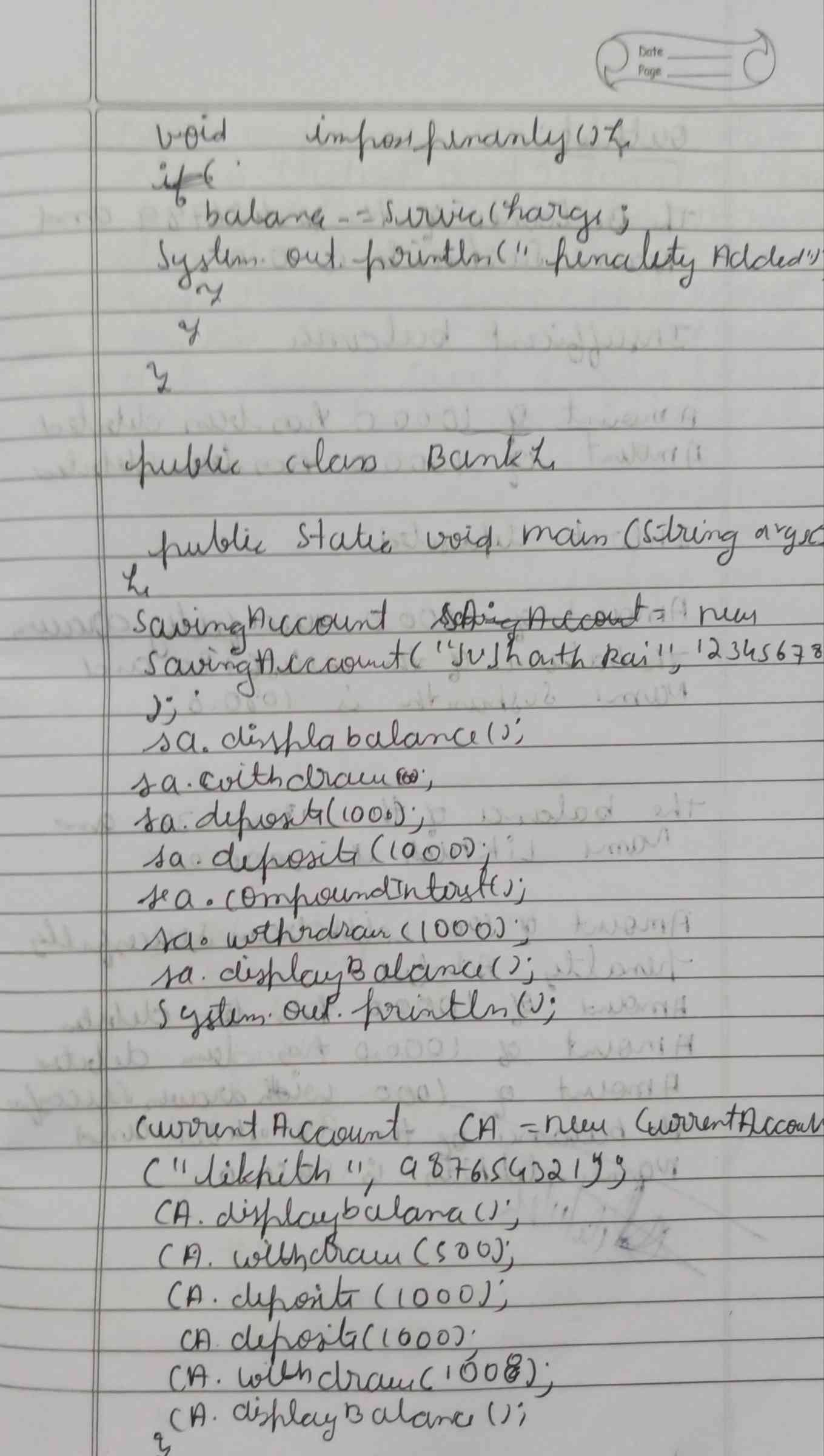
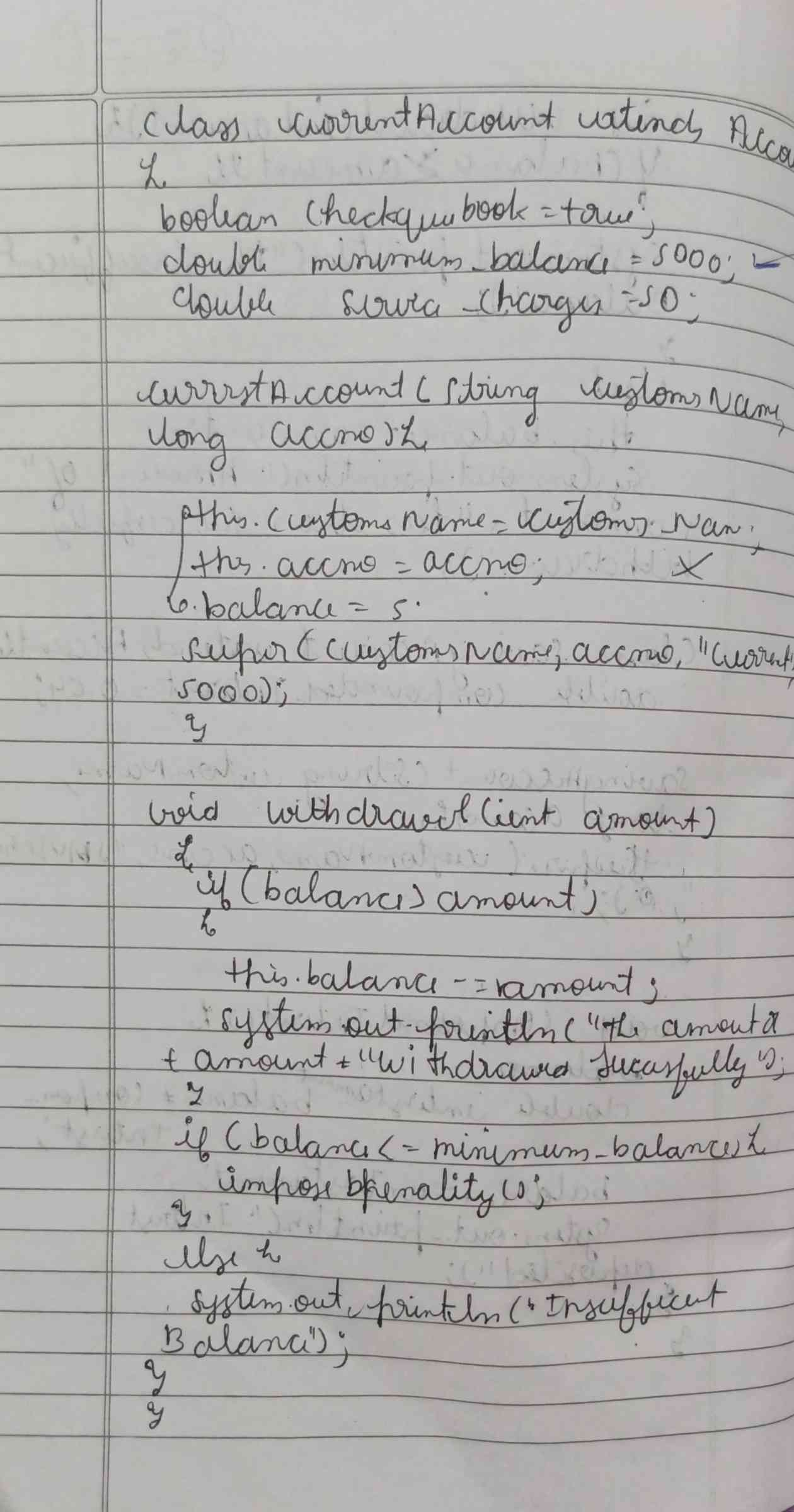
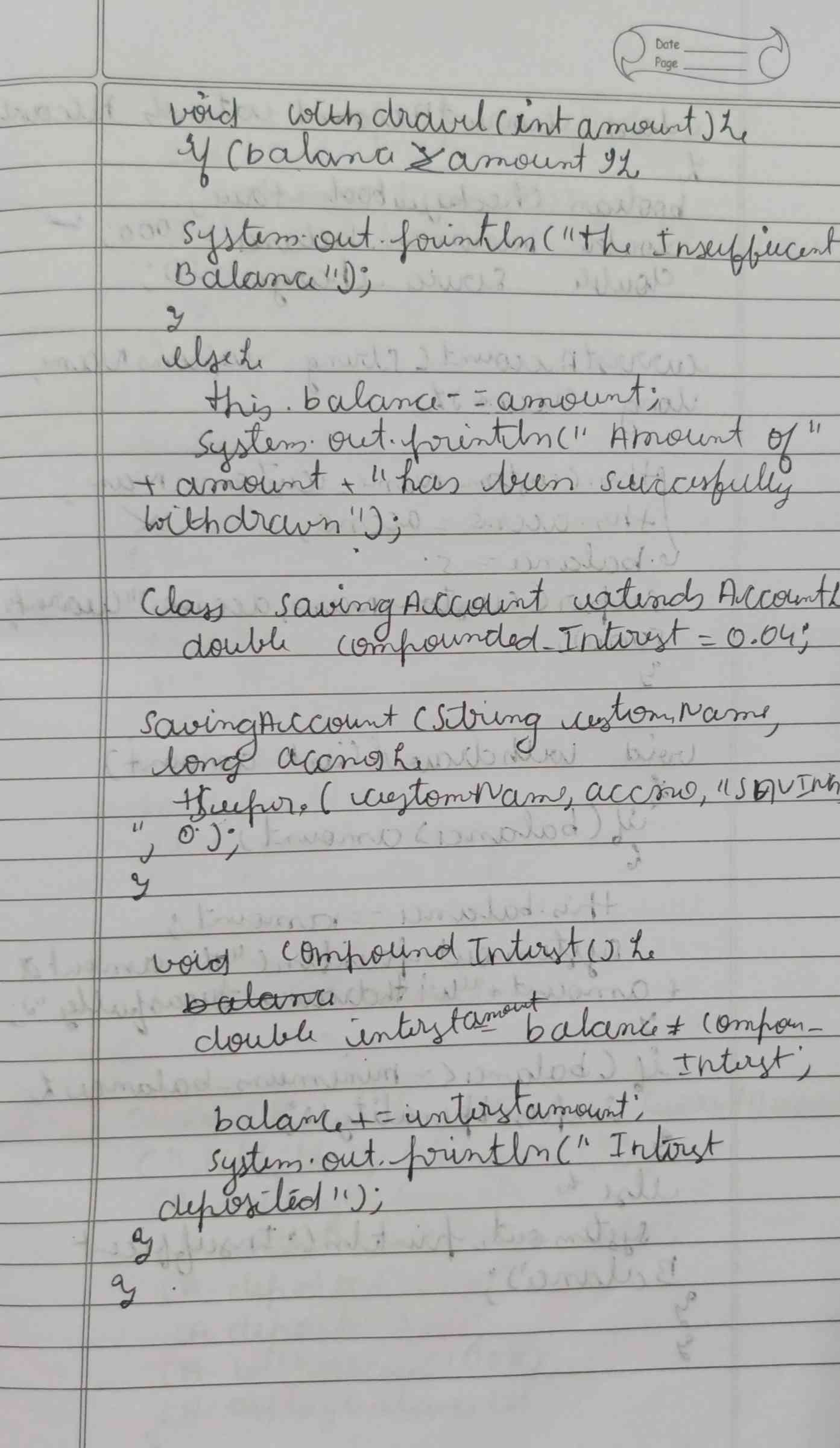
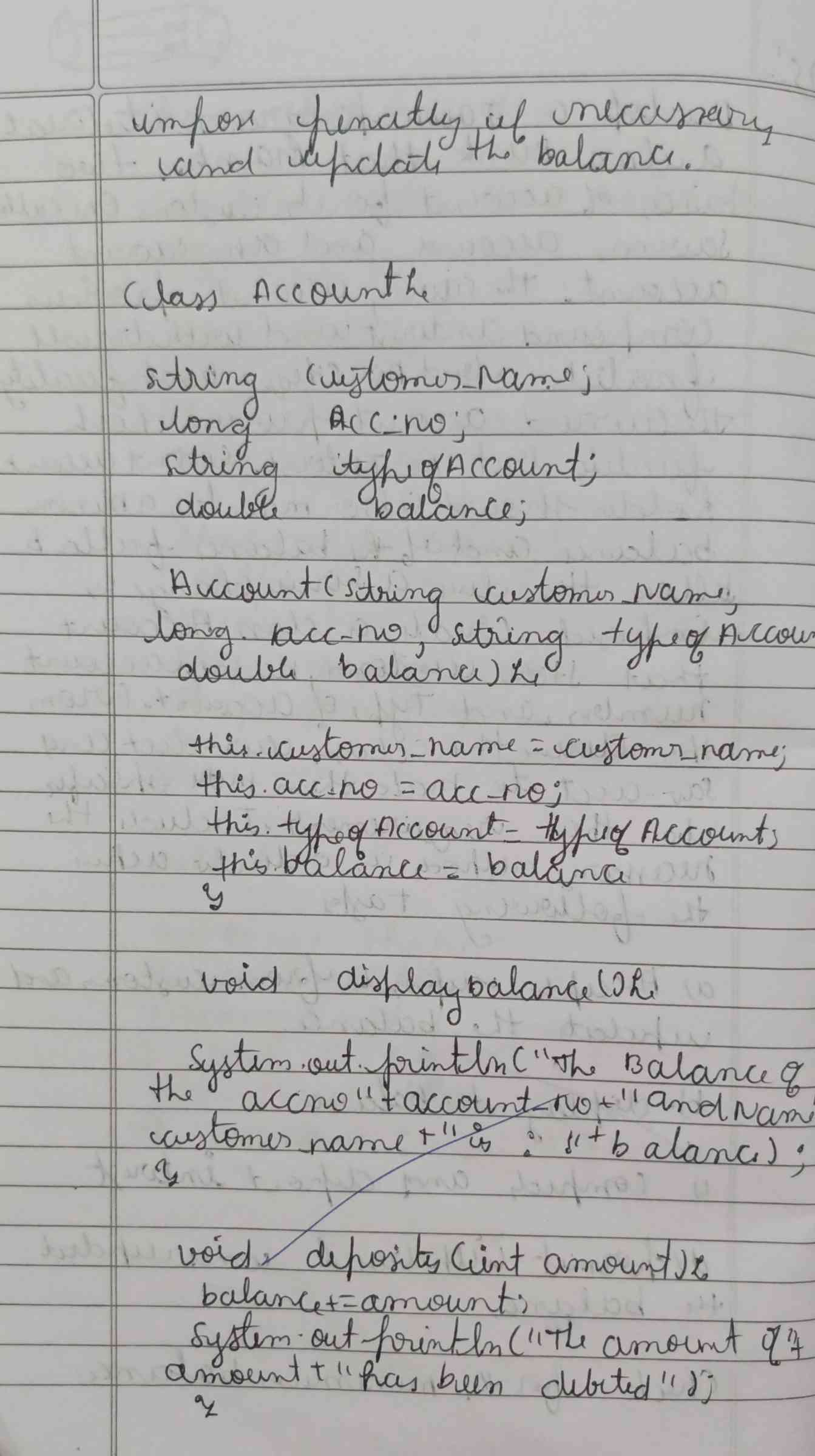
**LABORATORY PROGRAM – 5**

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: a) Accept deposit from customer and update the balance. b) Display the balance. c) Compute and deposit interest d) Permit withdrawal and update the balance Check for the minimum balance, impose penalty if necessary and update the balance.

**OBSERVATION :**

****



**CODE :**

**class Account{**

**String customer\_name;**

**long account\_no;**

**String typeofAccount;**

**double balance;**

**Account(String customer\_name,long account\_no,String typeAccount,double balance){**

**this.customer\_name=customer\_name;**

**this.account\_no=account\_no;**

**this.typeofAccount=typeAccount;**

**this.balance=balance;    }**

**void deposites(double amount){**

**this.balance+=amount;**

**System.out.println("Amount of "+amount+" has been debited");  }**

**void displayBalance(){**

**System.out.println("The Balance Of The "+account\_no+" and Name "+customer\_name+" is :"+balance);    }**

**void withdraw(double amount){**

**if(balance<amount){**

**System.out.println("Insufficient Balance");    }else{**

**this.balance-=amount;**

**System.out.println("Amount of "+amount+" succesfully withdrwn");}}}**

**class SavingAccount extends Account{**

**double compound\_interest=0.04;**

**SavingAccount(String customername,long account\_no){**

**super(customername,account\_no,"SAVINGS",0);}**

**void compoundInterest(){**

**double interest=balance\*0.04;**

**balance+=interest;**

**System.out.println("Intereset deposited");}  }**

**class CurrentAccount extends Account{**

**boolean chequebook=true;**

**double minimum\_Balance=5000;**

**double service\_charge=50;**

**CurrentAccount(String customername,long account\_no){**

**super(customername, account\_no,"CURRENT", 5000);  }**

**void withdraw(int amount){**

**if(balance>amount){**

**this.balance-=amount;**

**System.out.println("Amounte of "+amount+" withdrawed Succesfully");**

**imposePenalty();   }else{**

**System.out.println("Insuffescient  Balance");}}**

**void imposePenalty(){**

**if(balance<minimum\_Balance){**

**balance-=service\_charge;**

**System.out.println("Penalty Added");}    }  }**

**public class Bank {**

**public static void main(String[] args) {**

**SavingAccount savingAccount=new SavingAccount("sushanth",123456789 );**

**savingAccount.displayBalance();**

**savingAccount.withdraw(500);**

**savingAccount.deposites(1000);**

**savingAccount.deposites(1000);**

**savingAccount.compoundInterest();**

**savingAccount.withdraw(1000);**

**savingAccount.displayBalance();**

**System.out.println();**

**CurrentAccount currentAccount=new CurrentAccount("likhith",987654321 );**

**currentAccount.displayBalance();**

**currentAccount.withdraw(500);**

**currentAccount.deposites(1000);**

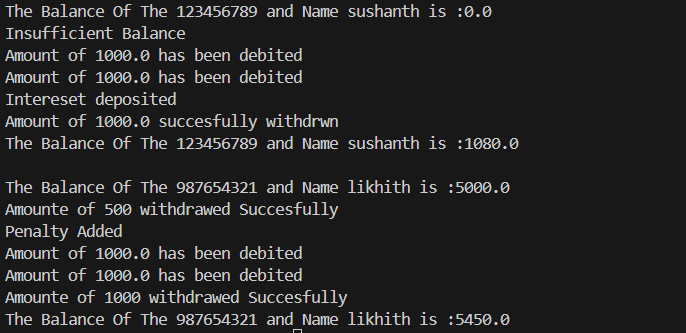
**currentAccount.deposites(1000);**

**currentAccount.withdraw(1000);**

**currentAccount.displayBalance(); }**

**}**

**OUTPUT :**

****