```
package A5;

import java.util.Scanner;

/**
 *
 * @author Lakshmi Priya
 */

class PANRequiredException extends Exception {
    private int pan;

    PANRequiredException(int pan) {
        this.pan = pan;
    }

    public String toString() {
        return "PAN Required" ;
    }
}
class MinBalRequiredException extends Exception {
    private float balance;

    MinBalRequiredException(float balance) {
        this.balance = balance;
    }

    public String toString() {
        return "Minimum Balance Required [ " + balance + " ]";
    }
}
class NoEnoughMoneyException extends Exception {
    private float balance, withdrawal;

    NoEnoughMoneyException(float balance, float withdrawal) {
        this.balance = balance;
        this.withdrawal = withdrawal;
    }

    public String toString() {
        return "No Enough Money [ Balance: "+ balance + " Withdrawal
request: "+withdrawal+" ]";
    }
}
class AccountNotFoundException extends Exception {
    private int accno;

    AccountNotFoundException(int accno) {
        this.accno = accno;
    }

    public String toString() {
        return "Account Not Found [ " + accno + " ]";
    }
}
```

```
class Account{
    private String cname;
    private int pan=0;
    private int accno;
    private String branch;
    private float balance;

    public Account(String cname, int accno, String branch, float
balance){
        this.cname=cname;
        this.accno=accno;
        this.branch=branch;
        this.balance=balance;
    }

    public Account(String cname, int pan, int accno, String branch,
float balance){
        this.cname=cname;
        this.pan=pan;
        this.accno=accno;
        this.branch=branch;
        this.balance=balance;
    }


    public String getCname(){
        return cname;
    }

    public int getPan(){
        return pan;
    }

    public int getAccno(){
        return accno;
    }

    public String getBranch(){
        return branch;
    }

    public float getBalance(){
        return balance;
    }

    public void setPan(int panno){
        pan=panno;
    }

    public void setBalance(float amt){
        balance=amt;
    }

    public static void deposit(int accno, float amt, Account acc[])
throws PANRequiredException{
        Account obj=null;
```

```
        try{
            obj=search(acc, accno);
        }
        catch(AccountNotFoundException e){
            System.out.println(e);
            return;
        }

        if(amt>50000 && obj.pan==0)
            throw new PANRequiredException(obj.getPan());

        obj.setBalance(obj.balance+amt);
        System.out.println("Amount credited successfully!");
        System.out.println("New Balance: Rs. "+obj.getBalance());
    }

    public static void withdraw(int accno, float amt, Account acc[])
throws NoEnoughMoneyException, MinBalRequiredException{
        Account obj=null;
        try{
            obj=search(acc, accno);
        }
        catch(AccountNotFoundException e){
            System.out.println(e);
            return;
        }

        if(amt>obj.balance)
            throw new NoEnoughMoneyException(obj.balance, amt);

        if(obj.balance-amt<1000)
            throw new MinBalRequiredException(obj.getBalance());

        obj.setBalance(obj.balance-amt);
        System.out.println("Amount debited successfully!");
        System.out.println("New Balance: Rs. "+obj.getBalance());
    }

    public static Account search(Account []acc, int accno) throws
AccountNotFoundException{
        int i;
        for(i=0;i<acc.length;i++){
            if(acc[i].getAccno()==accno)
                return acc[i];
        }
        if(i==acc.length)
            throw new AccountNotFoundException(accno);
        return null;
    }

    public static void displayAcc(Account []acc, int accno){
        Account obj=null;
        try{
            obj=search(acc, accno);
        }
        catch(AccountNotFoundException e){
```

```
            System.out.println(e);
            return;
        }

        System.out.println("\nCustomer name   : "+ obj.cname);
        System.out.println("Branch name     : "+obj.branch);
        System.out.println("Account number  : "+obj.accno);
        System.out.println("Account balance : "+obj.balance);
        if(obj.pan==0)
            System.out.println("PAN number not entered!");
        else
            System.out.println("PAN Number      : "+obj.pan);
    }

}

public class TestAccount {
    public static void main(String[] args) {
        String cname, branch;
        int pan, accno;
        float balance, amt;

        int n, i, choice=1;
        String flag;
        Scanner in=new Scanner(System.in);
        System.out.print("Enter number of records: ");
        n=in.nextInt();

        Account acc[]=new Account[n];
        Account obj=null;
        in.nextLine();
        for(i=0;i<n;i++)
        {
            System.out.print("\nEnter customer name   : ");
            cname=in.nextLine();
            System.out.print("Enter branch name     : ");
            branch=in.nextLine();
            System.out.print("Enter account number  : ");
            accno=in.nextInt();
            System.out.print("Enter account balance : ");
            balance=in.nextFloat();
            System.out.print("Want to enter PAN number? (y/n) : ");
            in.nextLine();
            flag=in.nextLine();
            if(flag.equalsIgnoreCase("y")){
                System.out.print("Enter PAN number      : ");
                pan=in.nextInt();
                acc[i]=new Account(cname, pan, accno, branch,
balance);
            }
            else
                acc[i]=new Account(cname, accno, branch, balance);

        }
```

```
        System.out.print("\nChoice:\n\t1. Deposit\n\t2.
Withdraw\n\t3. Set PAN\n\t4. Display account details\n\t0.
Exit\nEnter choice: ");
        choice=in.nextInt();
        while(choice!=0){

            switch(choice){
                case 1: System.out.print("Enter account number : ");
                        accno=in.nextInt();
                        System.out.print("Enter deposit amount : ");
                        amt=in.nextFloat();

                        try{
                            Account.deposit(accno, amt, acc);
                        }
                        catch(PANRequiredException e){
                            System.out.println(e);
                        }
                        break;

                case 2: System.out.print("Enter account number    :
");
                        accno=in.nextInt();
                        System.out.print("Enter withdrawal amount :
");
                        amt=in.nextFloat();

                        try{
                            Account.withdraw(accno, amt, acc);
                        }

catch(NoEnoughMoneyException|MinBalRequiredException e){
                            System.out.println(e);
                        }
                        break;

                case 3: System.out.print("Enter account number    :
");
                        accno=in.nextInt();
                        System.out.print("Enter PAN number      :
");
                        pan=in.nextInt();
                        try{
                            obj=Account.search(acc, accno);
                        }
                        catch(AccountNotFoundException e){
                            System.out.println(e);
                        }
                        if(obj!=null){
                            obj.setPan(pan);
                        }
                        break;

                case 4: System.out.print("Enter account number    :
");
                        accno=in.nextInt();
```

```
                        Account.displayAcc(acc, accno);
                        break;
                default: System.out.println("Invalid Choice!!");
            }
            System.out.print("\nChoice:\n\t1. Deposit\n\t2.
Withdraw\n\t3. Set PAN\n\t4. Display account details\n\t0.
Exit\nEnter choice: ");
            choice=in.nextInt();
        }

        System.out.println("\t\tCUSTOMER DETAILS");

        for(i=0;i<n;i++)
        {
            System.out.println("\nCustomer name   : "+
acc[i].getCname());
            System.out.println("Branch name     :
"+acc[i].getBranch());
            System.out.println("Account number  :
"+acc[i].getAccno());
            System.out.println("Account balance :
"+acc[i].getBalance());
            pan=acc[i].getPan();
            if(pan==0)
                System.out.println("PAN number not entered!");
            else
                System.out.println("PAN Number      : "+pan);

        }
    }
}
```