**INSTITUTE OF ENGINEERING &**

**MANAGEMENT**

**Department of Computer Science &**

**Engineering**



|  |  |
| --- | --- |
| **Name:** | Saptarshi Mondal |
| **Class Roll:** | 27 |
| **Enrollment No.:** | 12019002002039 |
| **Subject Name:** | **OOP Lab** |
| **Assignment No.:** | **Day 8** |
| **Date:** | **27/09/2021** |

import java.io.\*;

class Curr\_acct // CURRENT ACCOUNT CLASS

{

    final int max\_limit = 20;

    final int min\_limit = 1;

    final double min\_bal = 500;

    private String name[] = new String[20];

    private int accNo[] = new int[20];

    private String accType[] = new String[20];

    private double balAmt[] = new double[20];

    static int totRec = 0;

    // Intializing Methodpublicvoid initialize()

    {

        for (int i = 0; i < max\_limit; i++) {

            name[i] = "";

            accNo[i] = 0;

            accType[i] = "";

            balAmt[i] = 0.0;

        }

    }

    // TO ADD NEW RECORD

    public void newEntry() {

        String str;

        int acno;

        double amt;

        boolean permit;

        permit = true;

        if (totRec > max\_limit) {

            System.out.println("\n\n\nSorry we cannot admit you in our bank...\n\n\n");

            permit = false;

        }

        if (permit = true) // Allows to create new entry

        {

            totRec++; // Incrementing Total Record

            System.out.println("\n\n\n=====RECORDING NEW ENTRY=====");

            try {

                accNo[totRec] = totRec; // Created AutoNumber to accNo so no invalid id occurs

                System.out.println("Account Number :  " + accNo[totRec]);

                BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                System.out.print("Enter Name :  ");

                System.out.flush();

                name[totRec] = obj.readLine();

                accType[totRec] = "Current Account";

                System.out.println("Account Type : " + accType[totRec]);

                do {

                    System.out.print("Enter Initial  Amount to be deposited : ");

                    System.out.flush();

                    str = obj.readLine();

                    balAmt[totRec] = Double.parseDouble(str);

                } while (balAmt[totRec] < min\_bal); // Validation that minimun amount must be 500

                System.out.println("\n\n\n");

            } catch (Exception e) {

            }

        }

    }

    // TO DISPLAY DETAILS OF RECORD

    public void display() {

        String str;

        int acno = 0;

        boolean valid = true;

        System.out.println("\n\n=====DISPLAYING DETAILS OF CUSTOMER=====\n");

        try {

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account number : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.println("\n\nAccount Number : " + accNo[acno]);

                System.out.println("Name : " + name[acno]);

                System.out.println("Account Type : " + accType[acno]);

                System.out.println("Balance Amount : " + balAmt[acno] + "\n\n\n");

            }

        } catch (Exception e) {

        }

    }

    // TO DEPOSIT AN AMOUNT

    public void deposit() {

        String str;

        double amt;

        int acno;

        boolean valid = true;

        System.out.println("\n\n\n=====DEPOSIT AMOUNT=====");

        try {

            // Reading deposit value

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account No : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.print("Enter Amount you want to Deposit  : ");

                System.out.flush();

                str = obj.readLine();

                amt = Double.parseDouble(str);

                balAmt[acno] = balAmt[acno] + amt;

                // Displaying Depsit Details

                System.out.println("\nAfter Updation...");

                System.out.println("Account Number :  " + acno);

                System.out.println("Balance Amount :  " + balAmt[acno] + "\n\n\n");

            }

        } catch (Exception e) {

        }

    }

    // TO WITHDRAW BALANCE

    public void withdraw() {

        String str;

        double amt, checkamt, penalty;

        int acno;

        boolean valid = true;

        System.out.println("\n\n\n=====WITHDRAW AMOUNT=====");

        try {

            // Reading deposit value

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account No : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.println("Balance is : " + balAmt[acno]);

                System.out.print("Enter Amount you want to withdraw  : ");

                System.out.flush();

                str = obj.readLine();

                amt = Double.parseDouble(str);

                checkamt = balAmt[acno] - amt;

                if (checkamt >= min\_bal) {

                    balAmt[acno] = checkamt;

                    // Displaying Depsit Details

                    System.out.println("\nAfter Updation...");

                    System.out.println("Account Number :  " + acno);

                    System.out.println("Balance Amount :  " + balAmt[acno] + "\n\n\n");

                } else {

                    System.out.println("\n\nYour Balance has gone down and so penalty is calculated");

                    // Bank policy is to charge 20% on total difference of balAmt and min\_bal to be

                    // maintain

                    penalty = ((min\_bal - checkamt) \* 20) / 100;

                    balAmt[acno] = balAmt[acno] - (amt + penalty);

                    System.out.println("Now your balance revels : " + balAmt[acno] + "\n\n\n");

                }

            }

        } catch (Exception e) {

        }

    }

}

class Sav\_acct // SAVING ACCOUNT CLASS

{

    final int max\_limit = 20;

    final int min\_limit = 1;

    final double min\_bal = 1000;

    private String name[] = new String[20];

    private int accNo[] = new int[20];

    private String accType[] = new String[20];

    private double balAmt[] = new double[20];

    static int totRec = 0;

    // Intializing Methodpublicvoid initialize()

    {

        for (int i = 0; i < max\_limit; i++) {

            name[i] = "";

            accNo[i] = 0;

            accType[i] = "";

            balAmt[i] = 0.0;

        }

    }

    // TO ADD NEW RECORD

    public void newEntry() {

        String str;

        int acno;

        double amt;

        boolean permit;

        permit = true;

        if (totRec > max\_limit) {

            System.out.println("\n\n\nSorry we cannot admit you in our bank...\n\n\n");

            permit = false;

        }

        if (permit = true) // Allows to create new entry

        {

            totRec++; // Incrementing Total Record

            System.out.println("\n\n\n=====RECORDING NEW ENTRY=====");

            try {

                accNo[totRec] = totRec; // Created AutoNumber to accNo so no invalid id occurs

                System.out.println("Account Number :  " + accNo[totRec]);

                BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                System.out.print("Enter Name :  ");

                System.out.flush();

                name[totRec] = obj.readLine();

                accType[totRec] = "Saving Account";

                System.out.println("Account Type : " + accType[totRec]);

                do {

                    System.out.print("Enter Initial  Amount to be deposited : ");

                    System.out.flush();

                    str = obj.readLine();

                    balAmt[totRec] = Double.parseDouble(str);

                } while (balAmt[totRec] < min\_bal); // Validation that minimun amount must be 1000

                System.out.println("\n\n\n");

            } catch (Exception e) {

            }

        }

    }

    // TO DISPLAY DETAILS OF RECORD

    public void display() {

        String str;

        int acno = 0;

        boolean valid = true;

        System.out.println("\n\n=====DISPLAYING DETAILS OF CUSTOMER=====\n");

        try {

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account number : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.println("\n\nAccount Number : " + accNo[acno]);

                System.out.println("Name : " + name[acno]);

                System.out.println("Account Type : " + accType[acno]);

                // Bank policy is to give 10% interest on Net balance amt

                balAmt[acno] = balAmt[acno] + (balAmt[acno] / 10);

                System.out.println("Balance Amount : " + balAmt[acno] + "\n\n\n");

            }

        } catch (Exception e) {

        }

    }

    // TO DEPOSIT AN AMOUNT

    public void deposit() {

        String str;

        double amt;

        int acno;

        boolean valid = true;

        System.out.println("\n\n\n=====DEPOSIT AMOUNT=====");

        try {

            // Reading deposit value

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account No : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.print("Enter Amount you want to Deposit  : ");

                System.out.flush();

                str = obj.readLine();

                amt = Double.parseDouble(str);

                balAmt[acno] = balAmt[acno] + amt;

                // Displaying Depsit Details

                System.out.println("\nAfter Updation...");

                System.out.println("Account Number :  " + acno);

                System.out.println("Balance Amount :  " + balAmt[acno] + "\n\n\n");

            }

        } catch (Exception e) {

        }

    }

    // TO WITHDRAW BALANCE

    public void withdraw() {

        String str;

        double amt, checkamt;

        int acno;

        boolean valid = true;

        System.out.println("\n\n\n=====WITHDRAW AMOUNT=====");

        try {

            // Reading deposit value

            BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Enter Account No : ");

            System.out.flush();

            str = obj.readLine();

            acno = Integer.parseInt(str);

            if (acno < min\_limit || acno > totRec) // To check whether accNo is valid or Not

            {

                System.out.println("\n\n\nInvalid Account Number \n\n");

                valid = false;

            }

            if (valid == true) {

                System.out.println("Balance is : " + balAmt[acno]);

                System.out.print("Enter Amount you want to withdraw  : ");

                System.out.flush();

                str = obj.readLine();

                amt = Double.parseDouble(str);

                checkamt = balAmt[acno] - amt;

                if (checkamt >= min\_bal) {

                    balAmt[acno] = checkamt;

                    // Displaying Depsit Details

                    System.out.println("\nAfter Updation...");

                    System.out.println("Account Number :  " + acno);

                    System.out.println("Balance Amount :  " + balAmt[acno] + "\n\n\n");

                } else {

                    System.out.println("\n\nAs per Bank Rule you should maintain minimum balance of Rs 500\n\n\n");

                }

            }

        } catch (Exception e) {

        }

    }

}

class Bank {

    public static void main(String args[]) {

        String str;

        int choice, check\_acct = 1, quit = 0;

        choice = 0;

        Curr\_acct curr\_obj = new Curr\_acct();

        Sav\_acct sav\_obj = new Sav\_acct();

        System.out.println("\n=====WELLCOME TO BANK DEMO PROJECT=====\n");

        while (quit != 1) {

            try {

                BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                System.out.print("Type 1 for Current Account and Any no for Saving Account : ");

                System.out.flush();

                str = obj.readLine();

                check\_acct = Integer.parseInt(str);

            } catch (Exception e) {

            }

            if (check\_acct == 1) {

                do// For Current Account

                {

                    System.out.println("\n\nChoose Your Choices ...");

                    System.out.println("1) New Record Entry ");

                    System.out.println("2) Display Record Details ");

                    System.out.println("3) Deposit...");

                    System.out.println("4) Withdraw...");

                    System.out.println("5) Quit");

                    System.out.print("Enter your choice :  ");

                    System.out.flush();

                    try {

                        BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                        str = obj.readLine();

                        choice = Integer.parseInt(str);

                        switch (choice) {

                            case 1: // New Record Entry

                                curr\_obj.newEntry();

                                break;

                            case 2: // Displaying Record Details

                                curr\_obj.display();

                                break;

                            case 3: // Deposit...

                                curr\_obj.deposit();

                                break;

                            case 4: // Withdraw...

                                curr\_obj.withdraw();

                                break;

                            case 5:

                                System.out.println("\n\n.....Closing Current Account.....");

                                break;

                            default:

                                System.out.println("\nInvalid Choice \n\n");

                        }

                    } catch (Exception e) {

                    }

                } while (choice != 5);

            } else {

                do// For Saving Account

                {

                    System.out.println("Choose Your Choices ...");

                    System.out.println("1) New Record Entry ");

                    System.out.println("2) Display Record Details ");

                    System.out.println("3) Deposit...");

                    System.out.println("4) Withdraw...");

                    System.out.println("5) Quit");

                    System.out.print("Enter your choice :  ");

                    System.out.flush();

                    try {

                        BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                        str = obj.readLine();

                        choice = Integer.parseInt(str);

                        switch (choice) {

                            case 1: // New Record Entry

                                sav\_obj.newEntry();

                                break;

                            case 2: // Displaying Record Details

                                sav\_obj.display();

                                break;

                            case 3: // Deposit...

                                sav\_obj.deposit();

                                break;

                            case 4: // Withdraw...

                                sav\_obj.withdraw();

                                break;

                            case 5:

                                System.out.println("\n\n.....Closing Saving Account.....");

                                break;

                            default:

                                System.out.println("\nInvalid Choice \n\n");

                        }

                    } catch (Exception e) {

                    }

                } while (choice != 5);

            }

            try {

                BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

                System.out.print("\nEnter 1 for Exit : ");

                System.out.flush();

                str = obj.readLine();

                quit = Integer.parseInt(str);

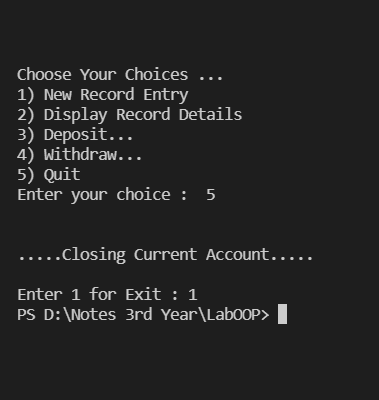
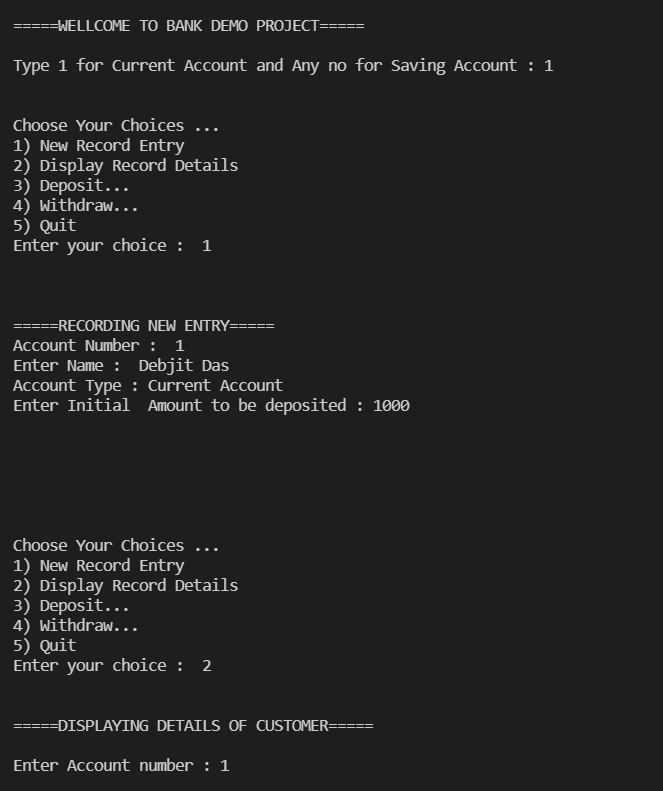
            } catch (Exception e) {

            }

        }

    }

}



class Subject {

    String code, title;

    int internal, theory, marks = 0;

    Subject(String code, String title, int internal, int theory) {

        this.code = code;

        this.title = title;

        this.internal = internal;

        this.theory = theory;

    }

}

class Student {

    int roll, total = 0, count = 0;

    String name, stream, college;

    Student(int roll, String name, String stream, String college, Subject... sub) {

        this.roll = roll;

        this.name = name;

        this.stream = stream;

        this.college = college;

        for (Subject s : sub) {

            s.marks = s.internal + s.theory;

            System.out.println("Subject code: " + s.code + ", Title: " + s.title + ", Marks: " + s.marks);

            total += s.marks;

            count++;

        }

    }

    @Override

    public String toString() {

        double avg = (double) total / (double) count;

        return "The student named " + name + " has roll number: " + roll + ", belongs to stream: " + stream + " of "

                + college + " college, has got a total of " + total + " marks and " + avg + " average marks.\n";

    }

}

public class Arguements {

    public static void main(String[] args) {

        System.out.println("Student Information");

        System.out.println("====================");

        System.out.println("Student 1: ");

        Student s1 = new Student(90, "Animesh Das", "ECE", "Techno", new Subject("ECE501", "Networking", 28, 60),

                new Subject("ECE503", "DBMS", 23, 53));

        System.out.println(s1);

        System.out.println("Student 2: ");

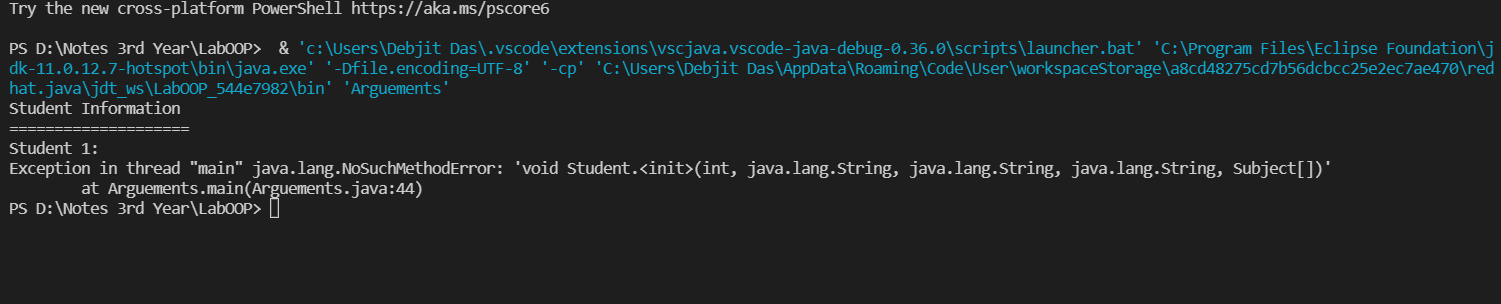
        Student s2 = new Student(106, "Abhinash Mehta", "CSE", "IEM", new Subject("CS501", "Ds Algo", 29, 65),

                new Subject("CS503", "DBMS", 25, 59));

        System.out.println(s2);

    }

}

class A {

    A() {

        System.out.println("Parent constructor executed.");

    }

    {

        System.out.println("Parent instance initialization block executed.");

    }

    static {

        System.out.println("Parent static block executed.");

    }

}

class B extends A {

    B() {

        System.out.println("Child constructor executed.");

    }

    {

        System.out.println("Child instance initialization block executed.");

    }

    static {

        System.out.println("Child static block executed.");

    }

}

public class InheritBlocks {

    public static void main(String[] args) {

        System.out.println();

        new B();

        System.out.println();

    }

}

