BANK

```
import java.util.Scanner;

class Account {     private String
customer_name;     private int
acc_no;     protected double
balance;

   public Account(String customer_name, int acc_no, double balance) {
this.customer_name = customer_name;
       this.acc_no = acc_no;
this.balance = balance;
   }

   public double getBalance() {
return balance;
   }

   public void deposit(double amount) {
      if (amount > 0) {
balance += amount;
          System.out.println("Deposited: " + amount);
      } else {
          System.out.println("Deposit amount must be positive.");
      }
   }
  public void withdraw(double amount)
    {
     if(amount<=getBalance()){          balance-
=amount;
        System.out.println("withdrew:"+amount + " balance is:"+ balance);      }
```

```
      else
        System.out.println("Insufficient funds!!");
    }
   public void displayBalance(){
      System.out.println("Current Balance: " + balance);
   }
}

class SavingsAccount extends Account {
private double interestRate;

   public SavingsAccount(String customerName, int accountNumber, double initialBalance, double interestRate) {       super(customerName, accountNumber, initialBalance);       this.interestRate = interestRate;
   }

   public void computeAndDepositInterest() {
double interest = getBalance() * interestRate / 100;
deposit(interest);
   }
}
class CurrentAccount extends Account {
private     double     minimumBalance;
private double serviceCharge;

   public CurrentAccount(String customerName, int accountNumber, double initialBalance, double minimumBalance, double serviceCharge) {       super(customerName, accountNumber, initialBalance);       this.minimumBalance = minimumBalance;       this.serviceCharge = serviceCharge;
   }
```

```
    public void checkMinimumBalance() {
    if (getBalance() < minimumBalance) {
            System.out.println("Balance is below minimum");           balance-
=serviceCharge;
            System.out.println("Deducted service charge:" +serviceCharge);
            System.out.println("Balance after deduction is:"+balance);
        }
    }
}

public class Bank {     public static void
main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("enter customer name:");
        String name=sc.nextLine();
System.out.println("enter accno:");        int
acc_no=sc.nextInt();
        System.out.println("enter initial balance:");
double balance=sc.nextDouble();
        System.out.println("enter minimum balance:");
double minimum_balance=sc.nextDouble();
System.out.println("enter interest rate:");        double
interest_rate=sc.nextDouble();
System.out.println("enter service charge:");
double service_charge=sc.nextDouble();
        System.out.println("Enter choice:\n 1.Current acc\n 2.Savings acc");
        int ch=sc.nextInt();
        System.out.println("Customer name is:"+ name+"\nAccount number:"+acc_no+"\nBhoomika
BG-1BM23CS067");

        switch(ch){
            case(1):
```

```
            System.out.println("account is current type");
            CurrentAccount ca = new
CurrentAccount(name,acc_no,balance,minimum_balance,service_charge);
           do{ System.out.println("enter choice:\n 1.deposit\n 2.withdraw\n 3.display balance");
            int c=sc.nextInt();
            ca.checkMinimumBalance();
            if(c==1){
              System.out.println("enter amount to be deposited:");
double amt=sc.nextDouble();
               ca.deposit(amt);}
else if(c==2){
              System.out.println("enter amount to withdraw:");
double amt=sc.nextDouble();                ca.withdraw(amt);}
else if(c==3){              ca.displayBalance();}
           else
            System.exit(0);
            }while(true);


        case(2):
            System.out.println("account is savings type");
           SavingsAccount sa=new SavingsAccount(name,acc_no,balance,interest_rate);
do{ System.out.println("enter choice:\n 1.deposit\n 2.withdraw\n 3.display balance");
           int c1=sc.nextInt();
if(c1==1){
              System.out.println("enter amount to be deposited:");
double amt=sc.nextDouble();
               sa.deposit(amt);}
           else if(c1==2){
              System.out.println("enter amount to withdraw:");
double amt=sc.nextDouble();            sa.withdraw(amt);}
else if(c1==3){
```

```
            sa.computeAndDepositInterest();
sa.displayBalance();}
            else{
             System.exit(0);
                }
            }while(true);
   }
}
```

```
C:\Users\Admin\Documents\23cs310>javac Bank.java

C:\Users\Admin\Documents\23cs310>java Bank
enter customer name:
Sharada
enter accno:
45982
enter initial balance:
25000
enter minimum balance:
5000
enter interest rate:
5
enter service charge:
100
Enter choice:
 1.Current acc
 2.Savings acc
1
Customer name is:Sharada
Account number:45982
Sharada
account is current type
enter choice:
 1.deposit
 2.withdraw
 3.display balance
1
enter amount to be deposited:
10000
Deposited: 10000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
2
enter amount to withdraw:
3000
withdrew:3000.0 balance is:32000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Current Balance: 32000.0
}
```

```
C:\Users\Admin\Documents\23cs310>java Bank
enter customer name:
Sharada\
enter accno:
467382
enter initial balance:
24000
enter minimum balance:
5000
enter interest rate:
5
enter service charge:
100
Enter choice:
 1.Current acc
 2.Savings acc
2
Customer name is:Sharada\
Account number:467382
Sharada
account is savings type
enter choice:
 1.deposit
 2.withdraw
 3.display balance
1
enter amount to be deposited:
2000
Deposited: 2000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
2
enter amount to withdraw:
500
withdrew:500.0 balance is:25500.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Deposited: 1275.0
Current Balance: 26775.0
```

PROGRAM 5

Develop a Java program to create a class Bank that maintains two kinds of accounts for its customers, one called savings and Current. The savings account provides compound interest and withdrawal facilities but no cheque book facilities. The current account provides cheque book facilities 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 class account that stores customer name, account no. & type of account. From this derive the class 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

Checks for min balance, impose penalty if necessary and update the balance.

```
import java.util.Scanner;

class Account {
    private String customer_name;
    private int acc_no;
    protected double balance;}

public Account(String customer_name, int acc_no, double bala
{ this.customer_name = customer_name;
  this.acc_no = acc_no;
  this.balance = balance;
}

public double getBalance() {
    return balance;
}

public void deposit(double Amount) {
    if (amount > 0) {
        balance += amount;
        S.O.P("Deposited:" + amount);
    }
    else {
    S.O.P("Deposit amount must be positive");}
}

public void withdraw(double amount) {
    if (amount <= getBalance()) {
        balance -= amount;
        S.O.P ("withdraw" + amount + "balance is" +
            balance);
    }
    else {
    S.O.P ("Insufficient balance") }

public void displayBalance() {
        S.O.P("Current balance" + balance);
    }
}
class SavingAccount extends Account {
    private double interestRate;

public SavingsAccount(String customername, int accountNumber,
                      double initialBalance, double interestRate) {
        this.interestRate = interestRate;
    }

public void computeAndDepositInterest() {
        double interest = getBalance() * interestRate / 100;
        deposit(interest);
    }
}

class currentAccount extends Account {
    private double minBalance;
    private double serviceCharge;

public CurrentAccount(String customerName, int accountNumber,
                      double minBalance, double serviceCha
    {
    super(customerName, accountName, initialBala
    this.minBalance = minBalance;
    this.serviceCharge= serviceCharge;
    }

public void checkMinBalance() {
    if (getBalance() < minBalance) {
    S.O.P("Balance is below minimum");
    balance -= serviceCharge;
    S.O.P(" Deducted service charge:" + serviceCharge)
    S.O.P("Balance after deduction is:" + Balance)
    }
}
}

public class Bank {
    PSVM(String[] args) {
    Scanner sc = new Scanner(System.in);
    S.O.P("Enter customer name");
    String name = sc.nextLine();
    S.O.P("Enter accno");
    int accno = sc.nextInt();
    SOP("enter initial balance");
    double balance = sc.nextDouble();
    S.O.P("enter minimum balance");
    double minBalance = sc.nextDouble;
    S.O.P("enter interest rate")
    double interest_rate = sc.nextDouble();
    S.O.P("enter service charge");
    double service_charge = sc.nextDouble();
    S.O.P("Enter choice\n1. Current Acc.\n
                2. SavingsAcc");
    int ch = sc.nextInt();
    S.O.P("Customer name is" + name +
        "\n Account number" + acc_no
        + "\n Shreya");

Switch (ch) {
    case1: S.O.P("account is current type");
    CurrentAccount ca = new CurrentAccount
        (name, acc_no, balance, min_bala
        service(charge));

    do { S.O.P("Enter choice:\n 1. deposit\n
                2. withdraw\n
                3. display balance
```

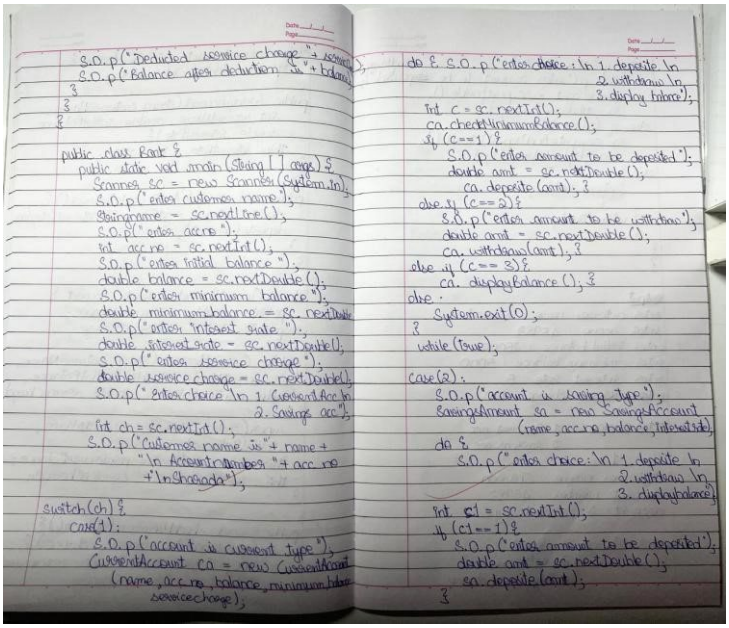
```
int c=sc.nextInt();
ca.checkMinBalance();
if(c==1){
    S.O.P("enter amount to be deposited");
    double amt=sc.nextDouble();
    ca.withdraw(amt);}
else if(c==3){
    ca.displayBalance();}
else{
    system.exit(0);
    }
    while(true);

case(2):
    S.O.P("account is saving type");
    savingsAcct sa=new SavingsAccount
                    (cname,accno,balance,interestrate);
    do{
    S.O.P("enter choice:\n 1.deposit\n
                2.withdraw\n
                3.displayBalance");

    int c1=sc.nextInt();
    if(c1==1){
        S.O.P("enter amt to be deposited");
        double amt=sc.nextDouble();
        sa.deposit(amt);
    }
    else if(c1==2){
        SOP("enter amt to be withdrawn");
        double amt=sc.nextDouble();
```

```
sa.withdraw(amt);}
else if(c1==3){
    sa.computeAndDepositInterest();
    sa.displayBalance();}
else{
    System.exit(0);
    }
    while(true);
    }
  }
}
```

Output

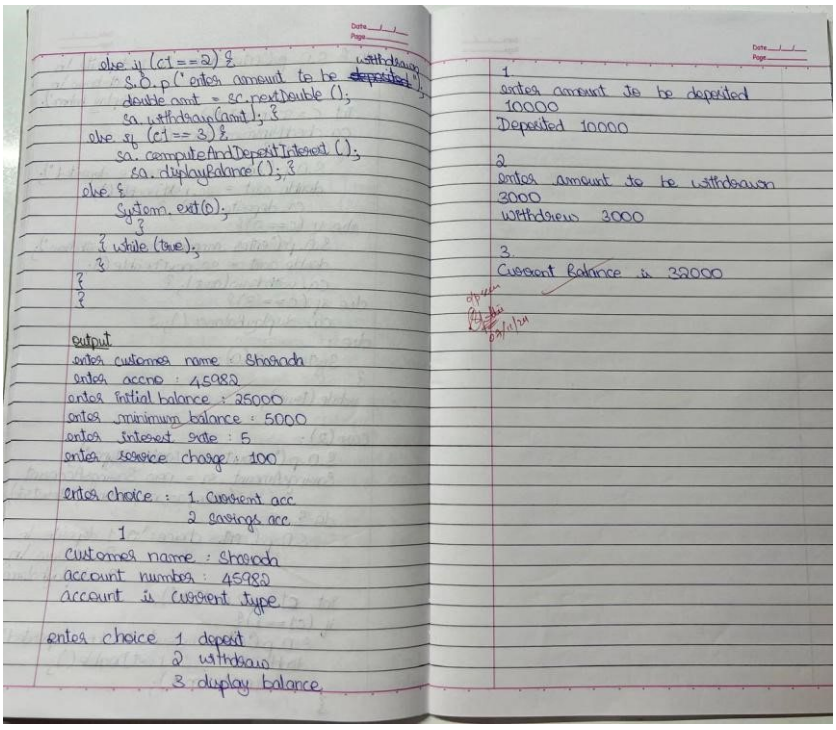
```
enter customer name: Shreya
enter accno: 45982
enter initial balance: 25000
enter min balance: 5000
enter interest rate: 5
enter service charge: 100

enter choice: 1. Current acc
              2. saving acc
1

customer name: Shreya
accno: 45982
account is current type

Enter choice  1 deposit
              2 withdraw
              3 display Balance
```

```
1
enter amt to be deposited
10000
Deposited 10000

2
enter amt to be withdrawn
3000
withdrew 3000

3
Current Balance is 32000
```



```
        S.O.p("Deducted service charge " + servicecharge);
        S.O.p("Balance after deduction " + balance);
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        S.O.p("enter customer name");
        String name = sc.nextLine();
        S.O.p("enter accno");
        int accno = sc.nextInt();
        S.O.p("enter initial balance");
        double balance = sc.nextDouble();
        S.O.p("enter minimum balance");
        double minimumbalance = sc.nextDouble
        S.O.p("enter interest rate");
        double interestrate = sc.nextDouble();
        S.O.p("enter service charge");
        double servicecharge = sc.nextDouble();
        S.O.p("enter choice \n 1. Current Acc \n
                2. Savings acc");
        int ch = sc.nextInt();
        S.O.p("Customer name is" + name +
            "\n Account number " + acc no
            + "\n Branch");

    switch(ch) {
        case(1):
            S.O.p("account is current type");
            CurrentAccount ca = new CurrentAccount
                (name, accno, balance, minimumbalance
                servicecharge);
```

```
            do { S.O.p("enter choice : \n 1. deposite \n
                                2. withdraw \n
                                3. display balance");
                int c = sc.nextInt();
                ca.checkMinimumBalance();
                if (c==1) {
                    S.O.p("enter amount to be deposited");
                    double amt = sc.nextDouble();
                    ca.deposite(amt);}
                else if (c==2) {
                    S.O.p("enter amount to be withdraw");
                    double amt = sc.nextDouble();
                    ca.withdraw(amt);}
                else if (c==3) {
                    ca.displayBalance();}
                else
                    System.exit(0);
                }
                while(true);

            case(2):
                S.O.p("account is saving type");
                SavingsAmount sa = new SavingsAccount
                        (name, accno, balance, interestrate);
                do {
                    S.O.p("enter choice: \n 1. deposite \n
                                2. withdraw \n
                                3. displaybalance");
                int c1 = sc.nextInt();
                if (c1==1) {
                    S.O.p("enter amount to be deposited");
                    double amt = sc.nextDouble();
                    sa.deposite(amt);
                }
```



```
            else if (c1==2) {                    withdraw
                S.O.p("enter amount to be ~~deposited~~");
                double amt = sc.nextDouble();
                sa.withdraw(amt);}
            else if (c1==3) {
                sa.computeAndDepositInterest();
                sa.displayBalance();}
            else {
                System.exit(0);
                }
            } while(true);
        }
    }
}
```

output

```
enter customer name : Sharada
enter accno : 45980
enter initial balance : 25000
enter minimum balance : 5000
enter interest rate : 5
enter service charge : 100

enter choice : 1. Current acc
               2. Savings acc
        1
customer name : Sharada
account number : 45980
account is current type

enter choice 1 deposit
             2 withdraw
             3 display balance
```

```
1
enter amount to be deposited
10000
Deposited 10000

2
enter amount to be withdrawn
3000
Withdrew 3000

3
Current Balance is 32000
```

dt