5) Develop a Java program to create a class Bank that maintains two kinds of accounts for its customers, one called Savings accounts and 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 Curr-acct and Sav-acct to make them more specific to their requirements. Include the following if necessary methods:

i) Accept deposit from customer and update the balance.
ii) Display balance
iii) Compute and deposit interest
iv) Permit withdrawal and update balance
v) Check for minimum balance, impose penalty if necessary and update the balance.

A.

```
import java.util.Scanner;
class Account
{
    private String name;
    private int accountNumber;
    private String type;
    public Account(String name, int accountNumber, String type)
    {
        this.name=name;
        this.accountNumber=accountNumber;
        this.type=type;
    }
```

```
public void setName(String name)
{   this.name = name;
}

public void setAccountNumber(int accountNumber)
{   this.accountNumber = accountNumber;
}

public void setType(String type)
{   this.type = type;
}

public String getName()
{
    return this.name;
}
public int getAccountNumber()
{   return this.accountNumber;
}
public String getType()
{
    return this.type;
}
}

class Curr_acct extends Account
{   private double balance;
    private boolean checkBook;
    private static double minBalance;
    private static double getServiceCharge();

    public static double getServiceCharge()
    {
        return serviceCharge;
    }
```

```
public Cour_acct (String name, int accountNumber, String type,
                  boolean checkBook) {
    super(name, accountNumber, type);
    this.checkBook = checkBook;
    this.balance = 0;
}

static {
    minBalance = 1000.00;
    serviceCharge = 5.00;
}

public double getBalance()
{   return this.balance;
}

public ~~double~~ void deposit(double amt)
{
    this.balance += amt;
}

public int withdraw(double amt)
{ if(this.balance - amt < minBalance && this.balance - amt > 0)
  {   this.balance -= serviceCharge * 0.01 * amt;
      this.balance -= amt;
      return 1;
}

~~this.balance >= amt~~;

if(this.balance - amt < 0)
  { return -1;
  }
  this.balance -= amt;
  return 2;
  }
}
```

```
class Sav_acct extends Account{
 private double balance;
 private static double interestRate;
 public Sav_acct(String name, int accountNumber, String type)
  {
    super(name, accountNumber, type);
    this.balance=0;
 }
 static{
       interestRate = 8.0;
 }
 public double getBalance(){
  {
    return this.balance;
 }
  public void deposit(double amt)
  {
    this.balance += amt;
 }
   public int withdraw(double amt){
    if(this.balance-amt<0)
    :{
       return -1;
    this balance -= amt;
    return 2;

public double calculate Interest()
 {
    double amt = (this.balance*(1.0+(InterestRate*0.1)));
    double interest = amt - this.balance;
    this.balance = amt;
    return interest;
  }
}
```

```
class BankMain{
public static void main(String[] args){
Scanner s = new Scanner(System.in);
System.out.println("Enter name");
String name = s.next();
System.out.println("Enter the account number");
int accountNumber = s.nextInt();
System.out.println("Enter the type");
System.out.println("1. Savings :");
System.out.println("2. Current :");
int type = s.nextInt();
if(type==2){
    System.out.println("Do u want a check book ?? y or n");
String checkBookString = s.next();
boolean checkBook;
if(checkBookString=="y")
   checkBook = true;
else
  checkBook = true;
else
  checkBook = false;
Curr_acct curr_acct = new Curr_acct(name, accountNumber,
                          "Current", checkBook);
int c;
do{
  displayMenu(false);
  c = s.nextInt();
  double amt;
  switch(c){
case 1:  System.out.println("The balance in account is"+curr_acct,
                                                  .getBalance());
      break;
```

```
case 2: System.out.println("Enter the amount to deposit");
        amt = s.nextDouble();
        curr_acct.deposit(amt);
        System.out.println("The balance in account is "
                                + curr_acct.getBalance());
        break;
case 3: System.out.println("Enter the amount to withdraw");
        amt = s.nextDouble();
        int exp = curr_acct.withdraw(amt);
        if(exp==1)
            System.out.println("Insufficient Balance");
    System.out.println("The balance in account is "+curr_acct.get
                                                    Balance());
        break;

case 5:
    break;
default : System.out.println("Please enter valid choice");

    }
}
while(c!=5);
}
else if(type==1)
    {
        Sav_acct sav_acct = new Sav_acct(name, accountNumber
                                        ,"Savings");
int c;
do {
    displayMenu(true);
    c = s.nextInt();
    double amt;
```

```
switch(c){
  case 1:  System.out.println(" The balance in account is"
                                  +sav_acct.getBalance());
           break;
  case 2: System.out.println(" Enter the amount to deposit");
          amt = s.nextDouble();
          sav_acct.deposit(amt);
          System.out.println(" The balance in account is "+sav_acct.get
                                                          Balance());
          break;
  case 3:
          System.out.println(" Enter the amount to withdraw ");
          amt = s.nextDouble();
          int exp = sav_acct.withdraw(amt);
          if(exp==-1)
             System.out.println(" Insufficient Balance ");
          System.out.println(" The balance in account is "+sav_acct.get
                                                          Balance());
          break;
  case 4:
          System.out.println(" The interest amount is "+sav_acct.
                                                  calculateInterest());
          System.out.println(" The balance in account is"+sav_acct.getBalance());
          break;
  case 5:  break;
  default : System.out.println(" Please enter valid choice ");
  }
}while(c!=5);
  }
}
public static void displayMenu(boolean isSavingAccount)
 {
  System.out.println(" 1. Check balance ");
  System.out.println(" 2. Deposit cash ");
```

```
        System.out.println("3. Withdraw Cash");
    if (isSavingsAccount)
        System.out.println("4. Calculate Interest");
    System.out.println("5. Exit");
    System.out.println("Enter your choice");
}
}
```

BankMain - Notepad

File Edit Format View Help

```
import java.util.Scanner;
class Account{
   private String name;
   private int accountNumber;
   private String type;
   public Account(String name,int accountNumber,String type){
      this.name=name;
      this.accountNumber=accountNumber;
      this.type=type;
   }
   public void setName(String name){
      this.name=name;
   }
   public void setAccountNumber(int accountNumber){
      this.accountNumber=accountNumber;
   }
   public void setType(String type){
      this.type=type;
   }
   public String getName(){
      return this.name;
   }
   public int getAccountNumber(){
      return this.accountNumber;
   }
   public String getType(){
      return this.type;
   }
}
class Curr  acct extends Account{
```

Ln 192, Col 2 | 100% | Windows (CRLF) | UTF-8

BankMain - Notepad

File Edit Format View Help

```
class Curr_acct extends Account{
   private double balance;
   private boolean checkBook;
   private static double minBalance;
   private static double serviceCharge;
   public static double getServiceCharge(){
      return serviceCharge;
   }
   public Curr_acct(String name,int accountNumber,String type,boolean checkBook){
      super(name,accountNumber,type);
      this.checkBook=checkBook;
      this.balance=0;
   }
   static{
      minBalance=1000.00;
      serviceCharge=5.00;
   }
   public double getBalance(){
      return this.balance;
   }
   public void deposit(double amt){
      this.balance+=amt;
   }
   public int withdraw(double amt){
      if(this.balance-amt<minBalance && this.balance-amt>0){
         this.balance-=serviceCharge*0.01*amt;
         this.balance-=amt;
         return 1;
      }if(this.balance-amt<0){
         return -1;
```

Ln 192, Col 2 | 100% | Windows (CRLF) | UTF-8

```
            return -1;
        }
        this.balance-=amt;
        return 2;
    }
}
class Sav_acct extends Account{
    private double balance;
    private static double interestRate;
    public Sav_acct(String name,int accountNumber,String type){
        super(name,accountNumber,type);
        this.balance=0;
    }
    static{
        interestRate=8.0;
    }
    public double getBalance(){
        return this.balance;
    }
    public void deposit(double amt){
        this.balance+=amt;
    }
    public int withdraw(double amt){
        if(this.balance-amt<0){
            return -1;
        }
        this.balance-=amt;
        return 2;
    }
    public double calculateInterest(){
```

BankMain - Notepad

File Edit Format View Help

```
    public double calculateInterest(){
        double amt=(this.balance*(1.0+(interestRate*0.01)));
        double interest=amt-this.balance;
        this.balance=amt;
        return interest;
    }
}
class BankMain{
    public static void main(String args[]){
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the name");
        String name=s.next();
        System.out.println("ENter the account number");
        int accountNumber=s.nextInt();
        System.out.println("Enter the type");
        System.out.println("1.Savings");
        System.out.println("2.Current");
        int type=s.nextInt();
        if(type==2){
            System.out.println("Do u want a check book ?? y or n");
            String checkBookString=s.next();
            boolean checkBook;
            if(checkBookString=="y")
                checkBook=true;
            else
                checkBook=false;
            Curr_acct curr_acct=new Curr_acct(name,accountNumber,"Current",checkBook);
            int c;
            do{
                displayMenu(false);
```

Ln 192, Col 2 | 100% | Windows (CRLF) | UTF-8

BankMain - Notepad

File Edit Format View Help

```
        do{
            displayMenu(false);
            c=s.nextInt();
            double amt;
            switch(c){
                case 1:
                    System.out.println("The balance in account is "+curr_acct.getBalance());
                    break;
                case 2:
                    System.out.println("Enter the amount to deposit");
                    amt=s.nextDouble();
                    curr_acct.deposit(amt);
                    System.out.println("The balance in account is "+curr_acct.getBalance());
                    break;
                case 3:
                    System.out.println("Enter the amount to withdraw");
                    amt=s.nextDouble();
                    int exp=curr_acct.withdraw(amt);
                    if(exp==1)
                        System.out.println("An service charge of "+(Curr_acct.getServiceCharge()*0.01*amt)+" was deducted");
                    else if(exp==-1)
                        System.out.println("Insufficent Balance");
                    System.out.println("The balance in account is "+curr_acct.getBalance());
                    break;
                case 5:
                    break;
                default:
                    System.out.println("Please enter valid choice");
            }
        }while(c!=5);
```

Ln 192, Col 2 | 100% | Windows (CRLF) | UTF-8

BankMain - Notepad

File Edit Format View Help

```
        }while(c!=5);
    }else if(type==1){
        Sav_acct sav_acct=new Sav_acct(name,accountNumber,"Savings");
        int c;
        do{
            displayMenu(true);
            c=s.nextInt();
            double amt;
            switch(c){
                case 1:
                    System.out.println("The balance in account is "+sav_acct.getBalance());
                    break;
                case 2:
                    System.out.println("Enter the amount to deposit");
                    amt=s.nextDouble();
                    sav_acct.deposit(amt);
                    System.out.println("The balance in account is "+sav_acct.getBalance());
                    break;
                case 3:
                    System.out.println("Enter the amount to withdraw");
                    amt=s.nextDouble();
                    int exp=sav_acct.withdraw(amt);
                    if(exp==-1)
                        System.out.println("Insufficent Balance");
                    System.out.println("The balance in account is "+sav_acct.getBalance());
                    break;
                case 4:
                    System.out.println("The interest amount is "+sav_acct.calculateInterest());
                    System.out.println("The balance in account is "+sav_acct.getBalance());
                    break;
```

BankMain - Notepad

File Edit Format View Help

```
                System.out.println("Enter the amount to withdraw");
                amt=s.nextDouble();
                int exp=sav_acct.withdraw(amt);
                if(exp==-1)
                   System.out.println("Insufficent Balance");
                System.out.println("The balance in account is "+sav_acct.getBalance());
                break;
            case 4:
                System.out.println("The interest amount is "+sav_acct.calculateInterest());
                System.out.println("The balance in account is "+sav_acct.getBalance());
                break;
            case 5:
                break;
            default:
                System.out.println("Please enter valid choice");
          }
        }while(c!=5);
    }
  }
  public static void displayMenu(boolean isSavingsAccount){
    System.out.println("1.Check balance");
    System.out.println("2.Deposit Cash");
    System.out.println("3.Withdraw Cash");
    if(isSavingsAccount)
       System.out.println("4.Calculate Interest");
    System.out.println("5.Exit");
    System.out.println("Enter your choice");
  }
}
```

Ln 192, Col 2 | 100% | Windows (CRLF) | UTF-8

Command Prompt

```
C:\Users\sohan\Desktop\Java Programs\bank>java BankMain
Enter the name
Sohan
ENter the account number
34213
Enter the type
1.Savings
2.Current
1
1.Check balance
2.Deposit Cash
3.Withdraw Cash
4.Calculate Interest
5.Exit
Enter your choice
2
Enter the amount to deposit
59999999
The balance in account is 5.9999999E7
1.Check balance
2.Deposit Cash
3.Withdraw Cash
4.Calculate Interest
5.Exit
Enter your choice
4
The interest amount is 4799999.920000002
The balance in account is 6.479999892E7
1.Check balance
2.Deposit Cash
3.Withdraw Cash
4.Calculate Interest
5.Exit
Enter your choice
3
Enter the amount to withdraw
5000
The balance in account is 6.479499892E7
1.Check balance
2.Deposit Cash
3.Withdraw Cash
4.Calculate Interest
5.Exit
Enter your choice
5

C:\Users\sohan\Desktop\Java Programs\bank>
```