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

```
import java.util.Scanner;

class account {
String customerName;
int accNum;
String typeAccount;
int balance=0;

Account( String customerName, int accountNum,
String typeaccount, int balance) {
this.customerName = customerName;
this.accountnum = accountnum;
this.typeaccount = typeaccount;
this.balance = balance;
}
void showbal(){
System.out.println(typeaccount + "balance is" +
balance);
}
void deposipt(){
int amount;
Scanner sc= new Scanner(System.in);
System.out.println("Enter amt to deposit");
amount = sc.nextInt();
balanceet =amt;
showbal();
}
void withdraw(amount){
if(amount > balance ){}
System.out.println("Insufficient");
}
else { balance -=amount;
System.out.print("The balance is" +balance);
}
```

.

```
boolean getcheque(){
    return false;
  }
}
class curracc extends account{
  int minbal=50000;
curracc(String customerName, int accountnum,
int balance){
  super(customerName, accountnum, "current",
    balance);
  }

void get(){
  Scanner sc = new Scanner(System.in);
  System.out.println("Enter customername,
  account number, and balance for current
  account:");
  customerName = sc.nextLine();
  accountnum = sc.nextInt();
  }
  void check(){
  if (balance < minbal)
    System.out.println("LOW balance");
    balance -=500;
  }
}

void withdraw(int amount){
  if(amount > balance){
  System.out.println("Insufficient");
  }
  else { balance -= amount;
    System.out.println("The balance:" +balance);
    check();
  }
}
```

```
boolean isCheque(){
    return false;
}

class saveacc extends account {
    int CI;
    floate rate = 7.0f;
    float years;

saveacc(String customerName, int accountnum,
int balance, float years){
    super(customername, int accountnum, "Saving",
      balance);
    this.years = years;
}
void get()
{
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter customer name,
    account number, balance and years of
    interest:");
    customerName = sc.nextLine();
    accountnum = sc.nextInt();
    balance = sc.nextInt();
    years = sc.nextFloat();
}
void cI(){
    CI = (int)(balance * Math.pow(1 + rate/100, years)
        - balance);
    balance += CI;
}
```

```
void ischeque()
{ System.out.println("Cheque available");
}

boolean ischeque() {
  return true
    }

}
class Banknew{
public static void main(String [] args){
  Scanner sc = new Scanner(System.in);
   int amt) ch, sch;
   boolean exit= false;

  While( ! exit){
System.out.println("Enter account type:
 1. Savings 2. Current (0 to exit): ");
  ch = sc.nextInt();
   if (ch ==0){
    exit = true;
    Continue;
    }
  switch(ch){
   case 1:
          Saveacc s = new Saveacc("",0,0);
          s.get();
```

withdraw 3. show 4. Check cheque 5. exit ");
int sch = sc.nextInt();
switch(sch)
{
case 1: c.deposit();
break;
case 2: System.out.println(" Enter amount to withdraw ");
amt = sc.nextInt();
c.withdraw(amt);
break;
case 3:
c.showbal();
break;
case 4: if(c.ischeque())
{ System.out.println(" Available ");}
case 5: case exit = true
break;
default: System.out.println("Invalid");
}
}
break;
default: System.out.println(" Invalid");
}

```
System.out.println("Enter choice: 1.deposit 2.
withdraw 3. show balance 4. compound Inter
5. check facility 6 Exit");
Sch = sc.nextInt();

switch (sch) {
case 1: s.deposit();
        break

case 2: System.out.println("Enter amt");
        amt = sc.nextInt();
        s.withdraw(amt);
        break;
case 3: showbal();
        break
case 4: s.CI();
        break;
case 5: if (s.isCheque()) {
            System.out.println("Cheque available");
        break;
case 6: savingsexit = true;
        break;
default: System.println("Invalid choice");
}
}
```

Enter account type 1.savings 2.current (0 to exit)

1

Enter customer name account number, balance and years for interest

Madhu

34000

2

Enter Choice 1. deposit 2.withdraw 3.show balance
4. Compound Interest

1

Enter amount to deposit

2300

Savings balance is : 36300

Enter choice 1.deposit 2.withdraw 3.show balance
4. compound Interest

4

compound interest : 1254

Balance after compound interest 41554

Enter choice : 1. Deposit 2. withdraw 3. show balance
4. compound interest

2

Enter amount to withdraw

50000

Insufficient balance

```
import java.util.Scanner;

class Account {
    String customerName;
    int accountNum;
    String typeAccount;
    int balance = 0;

    Account(String customerName, int accountNum, String typeAccount, int balance) {
        this.customerName = customerName;
        this.accountNum = accountNum;
        this.typeAccount = typeAccount;
        this.balance = balance;
    }

    void showBalance() {
        System.out.println(typeAccount + " balance is: " + balance);
    }

    void deposit() {
        int amount;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter amount to deposit: ");
        amount = sc.nextInt();
        balance += amount;
        showBalance();
    }
```

```
    void withdraw(int amount) {
        if (amount > balance) {
            System.out.println("Insufficient balance.");
        } else {
            balance -= amount;
            System.out.println("The balance is: " + balance);
        }
    }
}

class SaveAcc extends Account {
    int compoundInterest;
    float rate = 7.0f; // 7% interest
    float year;

    SaveAcc(String customerName, int accountNum, int balance, float year) {
        super(customerName, accountNum, "Savings", balance);
        this.year = year;
    }

    void get() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter customer name, account number, balance, and years for interest:
");
        customerName = sc.nextLine();
        accountNum = sc.nextInt();
        balance = sc.nextInt();
        year = sc.nextFloat();
    }
```

```
    void compoundInterest() {
        compoundInterest = (int)(balance * Math.pow((1 + rate / 100), year) - balance);
        balance += compoundInterest;
        System.out.println("Compound interest: " + compoundInterest);
        System.out.println("Balance after compound interest: " + balance);
    }
}

class CurrAcc extends Account {
    int minBalance = 50000;

    CurrAcc(String customerName, int accountNum, int balance) {
        super(customerName, accountNum, "Current", balance);
    }

    void get() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter customer name, account number, and balance for current
account: ");
        customerName = sc.nextLine();
        accountNum = sc.nextInt();
        balance = sc.nextInt();
    }

    void check() {
        if (balance < minBalance) {
            System.out.println("Low balance. Service charge of 500 imposed.");
            balance -= 500;
```

```
        }
    }

    void withdraw(int amount) {
        if (amount > balance) {
            System.out.println("Insufficient balance.");
        } else {
            balance -= amount;
            System.out.println("The balance is: " + balance);
            check();
        }
    }

    void isCheque() {
        System.out.println("Cheque facility is available.");
    }
}

public class newBank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int amount, Ch, c1;

        System.out.println("Enter account type 1. Savings 2. Current");
        Ch = sc.nextInt();

        while(true) {
            if (Ch == 1) {
```

```
            SaveAcc s = new SaveAcc("", 0, 0, 0);
            s.get();
            System.out.println("Enter choice 1. Deposit 2. Withdraw 3. Show balance 4.
Compound interest");
            c1 = sc.nextInt();
            if (c1 == 1) {
               s.deposit();
            } else if (c1 == 4) {
               s.compoundInterest();
            } else if (c1 == 2) {
               System.out.println("Enter amount to withdraw from savings account: ");
               amount = sc.nextInt();
               s.withdraw(amount);
            } else if (c1 == 3) {
               s.showBalance();
            } else {
               System.out.println("Invalid choice");
            }
         } else if (Ch == 2) {
            CurrAcc c = new CurrAcc("", 0, 0);
            c.get();
            System.out.println("Enter choice 1. Deposit 2. Withdraw 3. Show balance");
            c1 = sc.nextInt();
            if (c1 == 1) {
               c.deposit();
            } else if (c1 == 2) {
               System.out.println("Enter amount to withdraw from current account: ");
               amount = sc.nextInt();
               c.withdraw(amount);
```

```
            } else if (c1 == 3) {
                c.showBalance();
            } else {
                System.out.println("Invalid choice");
            }
        } else {
            System.out.println("Invalid choice");
        }

        System.out.println("Do you want to continue? (1. Yes / 0 or any chracter. No)");
        int continueChoice = sc.nextInt();
        if (continueChoice !=1) {
            break;
        }

    }

    sc.close();
}
```

```
}
Enter account type: 1. Savings 2. Current (0 to exit):
1
Enter customer name, account number, balance, and years for interest:
madhu
23
34000
2
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
3
Savings balance is: 34000
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
1
Enter amount to deposit:
2300
Savings balance is: 36300
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
4
Compound interest: 5259
Balance after compound interest: 41559
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
2
Enter amount to withdraw from savings account:
50000
Insufficient balance.
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
2
Enter amount to withdraw from savings account:
23000
The balance is: 18559
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
5
Cheque facility is not available for Savings account.
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Compound interest 5. Check Cheque facility 6. Exit
6
Enter account type: 1. Savings 2. Current (0 to exit):
2
Enter customer name, account number, and balance for current account:
vasisshta
345
120000
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Check Cheque facility 5. Exit
1
Enter amount to deposit:
24567
Current balance is: 144567
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Check Cheque facility 5. Exit
2
Enter amount to withdraw from current account:
90000
The balance is: 54567
```

```
The balance is: 24567
Low balance. Service charge of 500 imposed.
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Check Cheque facility 5. Exit
3
Current balance is: 24067
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Check Cheque facility 5. Exit
4
Cheque facility is available.
Enter choice: 1. Deposit 2. Withdraw 3. Show balance 4. Check Cheque facility 5. Exit
```