Write 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 saving account provides CI and withdrawal facilities but no cheque book facilit The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if balance falls below this level, a ervice charge is imposed.

Create a class Account that stores customer name, account number a type of account. From this deri the classes cur-acct and sav-acct to make them more specific to thier requirements. Include the necessary methods in order t achieve the following tasks:

) Accept deposit from customer & update the balance
) Display the balance
) Compute & deposit interest
) Permit withdrawal and update t balance. Check min balance impose p

```
import java.util.Scanner;
class Account
{
    String customerName;
    long accno;
    String accountType;
    double balance;
    public Account(String customerN
long accno, String accountType)
{
    this.customerName = customerNam
    this.accno = accno
    this.accountType = accountTyp
    this.balance = 0.0
}
public void displayBalance()
System.out.println("Account Numb
+ accno);
System.out.println("Customer Name:"
ustomerName);
System.out.println("Account Typ
accountType);
System.out.println("Balance:$"
balance);
```

```
class CurAcct extends Account {
    double minBalance;
    double serviceCharge;
    public CurAcct (String customerName,
long accNo)
{   super (customerName, accNo, "Current");
    this.minBalance = 500.0;
    this.serviceCharge = 50.0
}

public void withdraw (double amount)
{
    if (balance - amount >= minBalance)
        { balance -= amount; }
System.out.println ("Withdrawal
successful. Current Balance : $" + balance);
}
    else
{
  System.out.println ("Insufficient funds
\n Withdrawal not allowed.");
}

public void imposeServiceCharge ()
{
    if (balance < minBalance)
{   balance -= serviceCharge;
  System.out.println (" Servic charge
```

```
        emposed. Current Balance: Rs." + Balance);
    }
}
}
class SavAcct extends Account {
    double interestRate;
    Public SavAcct (String CustomerName,
long accno)
    { super (CustomerName, accno, "Saving");
      this. interestRate = 0.05
    }
    Public void compoundInterest (double
InitialAmount, intTerm)
    { double. CompoundInterest = initialamount
* Math.pow ((1 + interestRate), term) - initial
Amount;
    Balance += compoundInterest;
System. out. println (" compound Interest
deposited. Current Balance: Rs:" + balance);
}

ublic class Bank
{   public static void main (String [] 
args) {
    Scanner sc = new Scanner (System.in)
System.out.println ("Choose account type:
System.out.println ("1.Current 2.Saving"
```

```
System.out.println("Enter your choice:");
    int choice = sc.nextInt();
System.out.print("Enter customer name:");
String customerName = sc.next();
System.out.println("Enter account number:");
    long accNo = sc.nextLong();
if (choice == 1)
{ CurrAcct curAccount = new CurrAcct
(customerName, accNo);
    System.out.println("Enter intial balance: $");
    double initialBalance = sc.nextDouble();
curAccount.Balance = initialBalance;
System.out.print("Enter withdrawal amount: $");
    double withdrawalAmount = sc.nextDouble
curAccount.withdraw(withdrawalAmount);
    curAccount.imposeServiceCharge();
curAccount.displayBalance();
}
else if (choice == 2)
{
SavAcct savAccount = new SavAcct(cust
merName, accNo);
    System.out.print("Enter initial
balance: $");
```

```
double initialBalance = sc.nextDouble();
savAccount.balance = initialBalance;
System.out.print("Enter withdrawal amount: $");
double withdrawalAmount = sc.nextDouble();
savAccount.balance -= withdrawalAmount;
System.out.println("Withdrawal Successful. Current Balance: $" + savAccount.balance);
System.out.print("Enter interest Rate: ");
double InterestRate = sc.nextDouble();
savAccount.InterestRate = interestRate;
savAccount.displayBalance();
System.out.println("Enter term(in years) for CI Calculation: ");
int term = sc.nextInt();
savAccount.compoundInterest(initialBalance, term);
savAccount.displayBalance();
} else
{ System.out.println("Invalid choice");
}
```

Algorithm

Step1: Start

Step2: Initialise variables cust_name, accno, acc_type, balance

Step3: Input: enter customer name, accno balance account type as savings or current from user

Step4: enter the choice of savings & current account

Step5: Read the choice if choice is savings then step6 for current step7

Step6: enter intial balance, withdrawal amount & min balance from user check condtn for min balance i.e intial_balance - withdrawal should be >= Min i.e 500 then new Balance = balance - withdrawal)

then print currentBalance enter intrest rate & time to calculate CI

CI = balance * power (1+ intrest rate, time - intial balance

Print CI & deposit with intrest

Step7: else if (choice==2)

create obj of sav Acct initialize intialbalance, withdrawalamnt, intrest rate

Step 8: Stop

Proceed 19/11

```
C:\Users\bmsce\Desktop\1bm22cs027 ooj>javac Bank.java

C:\Users\bmsce\Desktop\1bm22cs027 ooj>java Bank
AKANKSHA SINGA
1BM22CS027
Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 1
Enter customer name: Akanksha
Enter account number: 027027027
Enter initial balance: $10000
Enter withdrawal amount: $5000
Withdrawal successful. Current Balance: $5000.0
Account Number: 27027027
Customer Name: Akanksha
Account Type: Current
Balance: $5000.0

C:\Users\bmsce\Desktop\1bm22cs027 ooj>javac Bank.java

C:\Users\bmsce\Desktop\1bm22cs027 ooj>java Bank
AKANKSHA SINGA
1BM22CS027
Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 2
Enter customer name: Akanksha
Enter account number: 027027027
Enter initial balance: $5000
Enter withdrawal amount: $500
Withdrawal successful. Current Balance: $4500.0
Enter interest rate: 0.05
Account Number: 27027027
Customer Name: Akanksha
Account Type: Savings
Balance: $4500.0
Enter term (in years) for compound interest calculation: 2
Compound Interest deposited. Current Balance: Rs.5012.5
Account Number: 27027027
Customer Name: Akanksha
Account Type: Savings
Balance: $5012.5
```

```
C:\Users\bmsce\Desktop\1bm22cs027 ooj>javac Bank.java

C:\Users\bmsce\Desktop\1bm22cs027 ooj>java Bank
AKANKSHA SINGA
1BM22CS027
Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 1
Enter customer name: Akanksha
Enter account number: 027027027
Enter initial balance: $800
Enter withdrawal amount: $400
Insufficient funds. Withdrawal not allowed.
Account Number: 27027027
Customer Name: Akanksha
Account Type: Current
Balance: $800.0

C:\Users\bmsce\Desktop\1bm22cs027 ooj>javac Bank.java

C:\Users\bmsce\Desktop\1bm22cs027 ooj>java Bank
AKANKSHA SINGA
1BM22CS027
Choose account type:
1. Current
2. Savings
Enter choice (1 or 2): 1
Enter customer name: Akanksha
Enter account number: 027027027
Enter initial balance: $400
Enter withdrawal amount: $100
Insufficient funds. Withdrawal not allowed.
Service charge imposed. Current Balance: Rs.350.0
Account Number: 27027027
Customer Name: Akanksha
Account Type: Current
Balance: $350.0
```