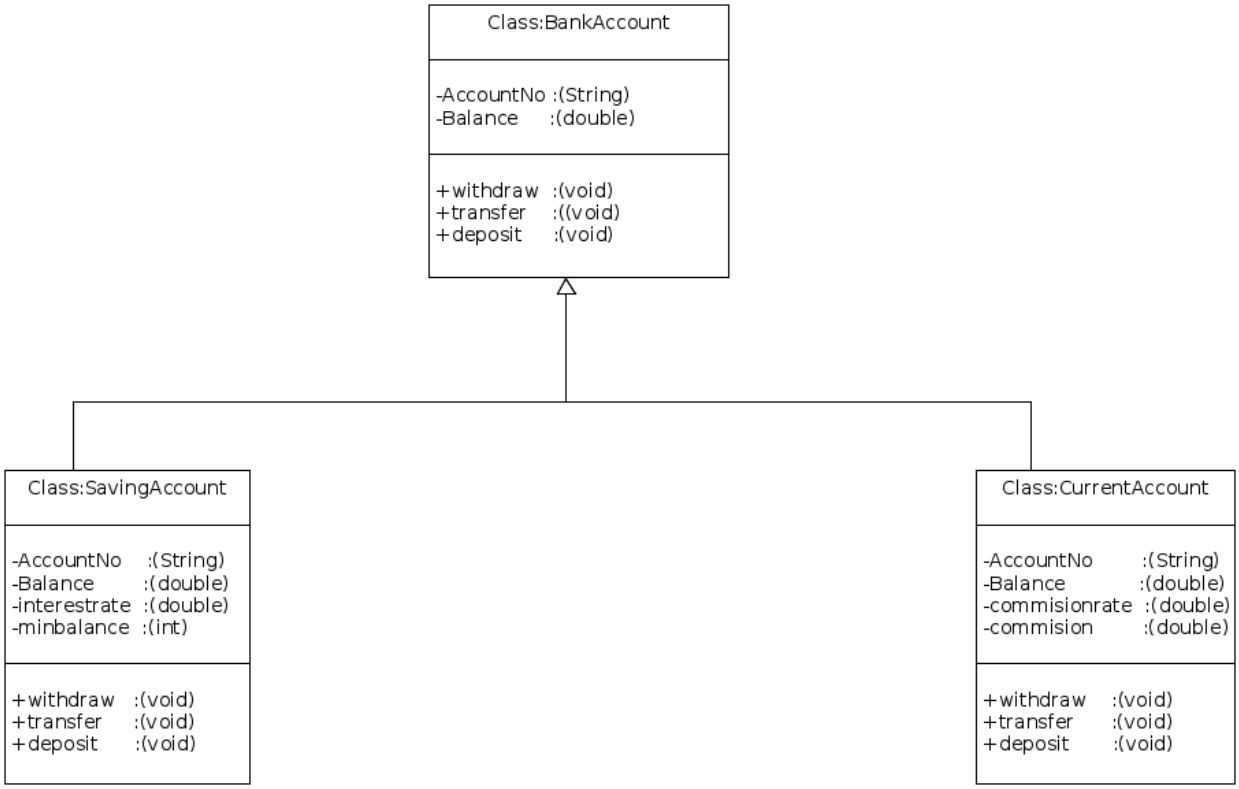
STATEMENT:

A bank account has an account number and a balance. The savings account is a bank account. It has an interest rate. A current bank account is a bank account. It has a commission rate. One can withdraw amount, deposit amount and transfer amount from any account. The savings bank account can withdraw amount with a minimum balance of ₹500. A current account will have a commission rate of 1% of the withdrawal amount. Create 2 savings account and 3 current accounts. Deposit amount in all the savings accounts. Withdraw amount from all the savings account. Transfer amount from one savings account to a current account. Create 3 current accounts, show withdrawals from all 3 current accounts and transfer amount from one current account to a savings account.

UMLET DIAGRAM:



CODE:

// PACKAGE MODEL :BANKACCOUNT CLASS

```
package model;

public class BankAccount {
    private String AccountNo;
    protected double balance;

    public BankAccount(){
        System.out.println("In BankAccount no argument
constructor"+"\n");
    }
```

```
    public String getAccountNo() {
        return AccountNo;
    }

    public double getBalance() {
        return balance;
    }

    public void setBalance(double balance) {
        this.balance = balance;
    }

    public BankAccount(String accountNo, double balance)
{
        super();
        AccountNo = accountNo;
        this.balance = balance;
    }

    public void Deposit(double amount) {
        this.balance = balance + amount;
        System.out.println(
                "AN AMOUNT OF  " + amount + " HAS BEEN
DEPOSITED IN THE ACCOUNT TO " + this.AccountNo+"\n");

    }

    public void withdraw(double amount) {
        this.balance = balance - amount;
        System.out.println("AN AMOUNT OF  " + amount + "
HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE " +
this.balance+"\n");
    }

    public void transfer(double amount, BankAccount
toAccount) {
        this.withdraw(amount);
        toAccount.Deposit(amount);
        System.out.println("THE" + "GIVEN AMOUNT IS
TRANSFER TO THE ACCOUNT  " + this.AccountNo+"\n");
    }


}
```

//PACKAGE MODEL :SUPERCLASS BANKACCOUNT:SUBCLASS:SAVINGACCOUNT

```
package model;

public class SavingAcc extends BankAccount {


    private static  int minbalance=500;
    private double interestrate;

    public double getInterestrate() {
        return interestrate;
    }

    public SavingAcc()
    {
        System.out.println("NO CONSTRUCTOR CALL");
    }
    public SavingAcc(String AccountNo , double balance ,
double interestrate) {
        super(AccountNo,balance);
        System.out.println("3 CONSTRUCTOR CALL");
        this.interestrate = interestrate;
    }

    public void withdraw(String AccountNo,double amount)
    {
        double s;
        double c=this.getBalance();
        s=this.getBalance()-amount;
        if(s<500)
        {
            System.out.println("YOU DON'T HAVE
SUFFICIENT CASH IN THE ACCOUNT");
        }
        else
        {
            c = c-amount;
            System.out.println("THE AMOUNT AVAILABLE IN
THE ACCOUNT IS : "+this.getBalance());
        }


    }
}
```

//PACKAGE MODEL : SUPERCLASS BANKACCOUNT :SUBCLASS CURRENTACCOUNT

```
package model;

public class CurrentAcc extends BankAccount {


    private static double commission=0.01;
    private double commissionrate;
    public double getCommissionrate() {
        return commissionrate;
    }
    public CurrentAcc(String AccountNo,double
balance,double commissionrate) {
        super(AccountNo,balance);
        System.out.println("ARGUMENT CONSTRUCTOR\n");
        this.commissionrate = commissionrate;
    }

    public void withdraw(double amount,String
AccountNo,double commision)
    {
        double s;
        double c=this.getBalance();

        //c=c-(c*commission);
        balance=balance-(amount+(balance*0.01));
        System.out.println("THE AMOUNT IN ACCOUNT IS :
"+ this.getBalance()+"\n");


    }
}
```

//PACKAGE USEMODEL : TESTBANKACCOUNT CLASS

```
package usemodel;

import model.CurrentAcc;
import model.SavingAcc;

public class TestBankAccount {

    public static void main(String[] args) {


        SavingAcc s1 = new SavingAcc("KRU123",6000,3.0);
```

```
		System.out.println("THE ACCOUNT NUMBER IS : " +s1.getAccountNo()+" HAVING BALANCE  OF : "+s1.getBalance()+" WITH INTEREST RATE : "+s1.getInterestrate());

		SavingAcc s2=new SavingAcc("KRU000",2000,2.0);
		System.out.println("THE ACCOUNT NUMBER IS : " +s2.getAccountNo()+" HAVING BALANCE  OF : "+s2.getBalance()+" WITH INTEREST RATE : "+s2.getInterestrate());

		System.out.println("\n\nINFO ABOUT SAVINGS ACCOUNT");
		System.out.println("THE BALANCE IN ACCOUNT"+" "+s1.getAccountNo()+" IS "+s1.getBalance());
		System.out.println("THE BALANCE IN ACCOUNT"+" "+s2.getAccountNo()+" IS "+s2.getBalance()+"\n\n");

		System.out.println("OPERATIONS ON SAVINGS ACCOUNT\n");

		s1.withdraw(2500);
		s2.Deposit(400);
		s1.transfer(300,s2);



		CurrentAcc c1=new CurrentAcc("SAN",20000,0.01);
		CurrentAcc c2=new CurrentAcc("SAN1",30000,0.01);
		CurrentAcc c3=new CurrentAcc("SAN2",10000,0.01);

		System.out.println("\n\nINFO ABOUT CURRENT  ACCOUNT");
		System.out.println("THE BALANCE IN ACCOUNT"+" "+c1.getAccountNo()+" IS "+c1.getBalance());
		System.out.println("THE BALANCE IN ACCOUNT"+" "+c2.getAccountNo()+" IS "+c2.getBalance());
		System.out.println("THE BALANCE IN ACCOUNT"+" "+c3.getAccountNo()+" IS "+c3.getBalance()+"\n\n");

		System.out.println("OPERATIONS ON CURRENT ACCOUNT\n");

		c1.withdraw(500, "SAN",0.01);
		c1.getBalance();
		s1.transfer(300, c1);
		c2.transfer(300, s2);
		c2.transfer(3000, c3);
		System.out.println("THE BALANCE IN ACCOUNT"+" "+s1.getAccountNo()+" IS "+s1.getBalance()+"\n");
		System.out.println("THE BALANCE IN ACCOUNT"+" "+s2.getAccountNo()+" IS "+s2.getBalance()+"\n");

		System.out.println("THE BALANCE IN ACCOUNT"+" "+c1.getAccountNo()+" IS "+ c1.getBalance()+"\n");
		System.out.println("THE BALANCE IN ACCOUNT"+" "+c2.getAccountNo()+" IS "+ c2.getBalance()+"\n");      System.out.println("THE BALANCE IN ACCOUNT"+" "+c3.getAccountNo()+" IS "+ c3.getBalance());}}
```

OUTPUT:

```
3 CONSTRUCTOR CALL
THE ACCOUNT NUMBER IS : KRU123 HAVING BALANCE  OF : 6000.0 WITH INTEREST RATE :
3.0
3 CONSTRUCTOR CALL
THE ACCOUNT NUMBER IS : KRU000 HAVING BALANCE  OF : 2000.0 WITH INTEREST RATE :
2.0


INFO ABOUT SAVINGS ACCOUNT
THE BALANCE IN ACCOUNT KRU123 IS 6000.0
THE BALANCE IN ACCOUNT KRU000 IS 2000.0


OPERATIONS ON SAVINGS ACCOUNT

AN AMOUNT OF  2500.0 HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE 3500.0

AN AMOUNT OF  400.0 HAS BEEN DEPOSITED IN THE ACCOUNT TO KRU000

AN AMOUNT OF  300.0 HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE 3200.0

AN AMOUNT OF  300.0 HAS BEEN DEPOSITED IN THE ACCOUNT TO KRU000

THEGIVEN AMOUNT IS TRANSFER TO THE ACCOUNT  KRU123

ARGUMENT CONSTRUCTOR

ARGUMENT CONSTRUCTOR

ARGUMENT CONSTRUCTOR



INFO ABOUT CURRENT  ACCOUNT
THE BALANCE IN ACCOUNT SAN IS 20000.0
THE BALANCE IN ACCOUNT SAN1 IS 30000.0
THE BALANCE IN ACCOUNT SAN2 IS 10000.0


OPERATIONS ON CURRENT ACCOUNT

THE AMOUNT IN ACCOUNT IS : 19300.0

AN AMOUNT OF  300.0 HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE 2900.0

AN AMOUNT OF  300.0 HAS BEEN DEPOSITED IN THE ACCOUNT TO SAN

THEGIVEN AMOUNT IS TRANSFER TO THE ACCOUNT  KRU123

AN AMOUNT OF  300.0 HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE 29700.0

AN AMOUNT OF  300.0 HAS BEEN DEPOSITED IN THE ACCOUNT TO KRU000

THEGIVEN AMOUNT IS TRANSFER TO THE ACCOUNT  SAN1

AN AMOUNT OF  3000.0 HAS BEEN WITHDRAWN FROM ACCOUNT CURRENT BALANCE 26700.0

AN AMOUNT OF  3000.0 HAS BEEN DEPOSITED IN THE ACCOUNT TO SAN2

THEGIVEN AMOUNT IS TRANSFER TO THE ACCOUNT  SAN1
```

THE BALANCE IN ACCOUNT KRU123 IS 2900.0

THE BALANCE IN ACCOUNT KRU000 IS 3000.0

THE BALANCE IN ACCOUNT SAN IS 19600.0

THE BALANCE IN ACCOUNT SAN1 IS 26700.0

THE BALANCE IN ACCOUNT SAN2 IS 13000.0