5. Develop a Java program to create a class Bank that maintains two kinds of accounts 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 cu-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

```
class Accounts
{
    String customer_name;
    long acc_no;
    String typeofAccount;
    double balance;
    Accounts ( String customer_name, long
    acc_no, String typeofAccount, double bal
    -ance){
        this.customer_name = customer_name;
        this.acc_no = acc_no;
        this.typeofAccount = typeofAccounts;
        this.balance = balance;
    }
```

```
void displaybalance()
{
System.out.println("The balance of the
accno"+account_no+"and name"+ customer
name+" is:"+balance);
}
void deposite(int amount)
{
balance+=amount;
System.out.Println("The amount of "+am
-ount+" has been debited");
}
void withdrawl(int amount){
if(balance > amount)
{
System.out.println("The Insuffient
Balance");
}
else{
this.balance -= amount;
System.out.println("Amount of "+
amount+" has been succesfully withdrawn");
}
}
class Saving_Account extends Account {
double compounded_Interest = 0.04;
savingAccount(String customername, long
acc_no)
{
super(customername, accno, "saving"
,0);
}
```

```
void compound_Interest()
{
  double interest_amount += balance * Compound_
  Interest;
  balance += interest amount;
  System.out.println("Interest deposited:");
  }
}

class currentAccount extends Account
{
  boolean checkbookk = True;
  double minimum_balance = 5000;
  double service_charges = 50;

currentAccount(String customername, long
  accno){

  super(customer name, accno, "current", 5000);
  }
void withdraw(int amount)
  {
    if(balance > amount)
    {
      this.balance -= amount;
      System.out.println("The amount of
    amount = " withdraw Succesfully");
    }
    if(balance <= minimum_balance)
    { 'impose penality();
    }
  else {
      System.out.Println(" Inssuffient Balance
```

```
void importPenalty(){
    balance -= service_charge;
    System.out.println("Penalty added");
  }
 }
}

Public class Bank{
 Public static void main(String args[])
 {
  SavingsAccount sa = new
  SavingAccount("Umesh", 12345678);
   sa.displaybalance();
   sa.withdraw();
   sa.deposit(1000);
   sa.deposit(1000);
   sa.compoundInterest();
   sa.withdraw(1000);
   sa.displaybalance();
   System.out.println();

  CurrentAccount CA = new CurrentAccount("
   Nikhil", 98765432);
   CA.displaybalance();
   CA.withdraw(500);
   CA.deposits(1000);
   CA.deposite(1000);
   CA.withdraw(1000);
   CA.displayBalance();

  }
 }
```

o/p

The Balance of the 12345678 & name umesh is 0.0

Insuffient balance

Amount of 1000.0 has been debited
Amount of 1000.0 has been debited

Interest deposited

Amount of 1000.0 succesfully drawn
The balance of the 12345678 and name umesh is 1080.6

The balance of the 98765432 and name nikhil is 5000.00

Amount of 500 withdrawn succesfully
Penalty added.
Amount of 1000.00 has been debited
Amount of 1000.00 has been debited
Amount of 1000 withdrawn succesfully
The balance of the 98765321 & name nikhil is 5450.0