*'''*

*Created on Apr. 29, 2020*

**@author:** *nirmal*

*@class: Bank.py*

*'''*

class **Bank**:

IFSC\_Code = 241

bankname = *"Standard Chartered Bank"*

def **\_\_init\_\_**(*self*, branchname, branchlocation):

*self*.branchname = branchname

*self*.loc = branchlocation

*'''*

*Created on Apr. 29, 2020*

**@author:** *nirmal*

*@class: Customer.py*

*'''*

class **Customer**():

def **\_\_init\_\_**(*self*, customerid, customername, customeraddress, details):

*self*.CustomerID = customerid

*self*.custname = customername

*self*.address = customeraddress

*self*.contactdetails = details

*'''*

*Created on Apr. 29, 2020*

**@author:** *nirmal*

*@class: Account.py*

*'''*

from Bank import Bank

class **Account**(Bank):

branchname = *"Jubilee Hills Corporate Office"*

branchlocation = *"Jubilee Hills"*

def **\_\_init\_\_**(*self*, accountID, customer, balance, SMinBalance):

*self*.AccountID = accountID

*self*.Cust = customer

*self*.balance = balance

*self*.SMinBalance = SMinBalance

super(Account, *self*).\_\_init\_\_(*self*.branchname, *self*.branchlocation)

def **getAccountInfo**(*self*):

print(*"Customer Name:"* + *self*.Cust.custname)

def **deposit**(*self*, depositAmount = 2000, state = *'true'*):

pass

def **withdraw**(*self*, withdrawalAmount = 500):

if( withdrawalAmount < *self*.balance - *self*.SMinBalance):

*self*.balance -= withdrawalAmount

print(*"Amount"* + str(withdrawalAmount) + *"withdrawn from account, new balance is :"* + str(*self*.balance))

else:

print(*"Amount to withdraw exceeds minimum balance"*)

def **getBalance**(*self*):

print(*"Current balance is: "* + str(*self*.balance))

*'''*

*Created on Apr. 29, 2020*

**@author:** *nirmal*

*@class: Account.py*

*'''*

from Account import Account

class **SavingsAccount**(Account):

balance = 40

SMinBalance = 10

def **\_\_init\_\_**(*self*, accountID, customer):

super(SavingsAccount, *self*).\_\_init\_\_(accountID, customer, *self*.balance, *self*.SMinBalance)

def **getSavingAccountInfo**(*self*):

print(*"Welcome, Customer Name:"* + *self*.Cust.custname)

def **deposit**(*self*, depositAmount, allowed = *'true'*):

if(allowed):

*self*.balance += depositAmount

def **withdraw**(*self*, withdrawalAmount):

if( withdrawalAmount > *self*.balance - *self*.SMinBalance):

print(*"Not sufficient balance"*)

else:

*self*.balance -= withdrawalAmount

print(*"Balance successfully withdrawn for Amount:"* + str(withdrawalAmount) + *". New balance is: "* + str(*self*.balance))

def **getBalance**(*self*):

print(*"Current Savings Account Balance is:"* + str(*self*.balance))

*'''*

*Created on Apr. 29, 2020*

**@author:** *nirmal*

*@class: Main.py*

*'''*

from Customer import Customer

from SavingsAcount import SavingsAccount

def **main**():

customerid = input(*"Please Enter your Customer ID: "*)#432

customername = input(*"Please Enter your Name: "*)#"Nirmal"

customeraddress = input(*"Please Enter you address: "*) #"Hyderabad"

customerdetails = input(*"Please Enter your details: "*) #"ComplexObject"

accountID = input(*"Please Enter your Account ID: "*) #781

customer = Customer(customerid, customername, customeraddress, customerdetails)

savingsaccount = SavingsAccount(accountID, customer)

savingsaccount.getSavingAccountInfo()

while(True):

transactionType = int(input(*"Please Enter 1 for getting Account Balance, 2 for Deposit, 3 for withdrawal, Any other key and Enter for logging out : \n"*))

if(transactionType == 1):

savingsaccount.getBalance()

elif(transactionType == 2):

amountToDeposit = int(input(*"Enter the amount to deposit: "*))

savingsaccount.deposit(amountToDeposit)

savingsaccount.getBalance()

elif(transactionType == 3):

amountToWithdraw = int(input(*"Enter withdrawal Amount:"*))

savingsaccount.withdraw(amountToWithdraw)

else:

print(*"Logging out!!"*)

break

if \_\_name\_\_ == *'\_\_main\_\_'*:

main()