

'''Implement a class called BankAccount that represents a bank account. The class should have private attributes for account number, account holder name, and account balance. Include methods to deposit money, withdraw money, and display the account balance. Ensure that the account balance cannot be accessed directly from outside the class. Write a program to create an instance of the BankAccount class and test the deposit and withdrawal functionality.'''

```
class BankAccount:

    def __init__(self, account_number, account_holder_name, initial_balance=0.0):

        self.__account_number = account_number

        self.__account_holder_name = account_holder_name

        self.__account_balance = initial_balance

    def deposit(self, amount):

        if amount > 0 :

            self.__account_balance += amount
```



```
        #self.__account_b  
alance = self.__account_ba  
alance+amount  
  
        print("Deposited ₹{  
}. New balance: ₹{}".forma  
t(amount, self.__account_b  
alance))  
  
    else:  
  
        print("Invalid dep  
osit amount. Please depos  
it a positive amount.")  
  
    def withdrawl(self, amou  
nt):  
  
        if amount > 0 and a  
mount <= self.__account_b  
alance:
```



```
        self.__account_balance -= amount

        #self.__account balance = self.__account_balance-amount

        print("Withdrew ₹{:. New balance: ₹{:.format(
amount, self.__account_balance))

    else:

        print("Invalid withdrawal amount or insufficient balance.")

    def display_balance(self):

        print("Account balance
```



```
for {} (Account #{}): ₹{}".f
ormat(self.__account_hold
er_name,

        self.__account_num
ber,

        self.__account_bala
nce))

#Create an instance of the
BankAccount class

account = BankAccount(ac
count_number="123456789
",

        account_hol
der_name="Saridha",

        initial_balan
ce=5000.0)
```



#Test deposit and withdraw
wl functionality

account.display_balance()

account.deposit(500.0)

account.withdrawl(200.0)

account.withdrawl(2000.0)

account.display_balance()



Catalogue



Settings