

# Sasta banking APP

This is a small prototype of a banking application that has been created just for assignment purposes. This is not a real banking application just seeks to replicate the core functionality of depositing and withdrawing amount of money.

Hence the prefix sasta which means cheap is used.

This app will make the use of OOPs of Python for working!

```
In [67]: #creating the account class
class Bank_account:
    def __init__(self,acc_no,holder_name,acc_type,balance):
        self.acc_no = acc_no
        self.holder_name = holder_name
        self.acc_type = acc_type
        self.balance=balance
    def createAccount(self):
        self.acc_no= int(input("Enter the account no : "))
        self.holder_name = input("Enter the account holder name : ")
        self.acc_type = input("Enter the type of account [C/S] : ")
        self.balance = int(input("Enter The Initial amount(>=500 for Saving and >=1000 for current)"))
        print("\n Account Created")
    def report(self):
        print(self.acc_no, " ",self.holder_name," ",self.acc_type," ", self.balance," ")
    def getAccountNo(self):
        return self.acc_no
    def getAcccountHolderName(self):
        return self.holder_name
    def getAccountType(self):
        return self.acc_type
    def getbalance(self):
        return self.balance
    def Account_display(self):
        print("Account Number : ",self.acc_no)
        print("Account Holder Name : ", self.holder_name)
        print("Type of Account",self.acc_type)
        print("Balance : ",self.balance)
    def Account_modify(self):
        print("Account Number : ",self.acc_no)
        self.holder_name = input("Modify Account Holder Name :")
        self.acc_type = input("Modify type of Account :")
        self.balance = int(input("Modify Balance :"))
    def depositAmount(self,depamt):
        depamt = float(input("Enter amount to be deposited: "))
        if depamt>0:
            self.balance += depamt
            print("\n Amount Deposited:", depamt)
            print("\n Amount total:", self.balance)
        else:
            print("enter some money!!!")
    def withdrawAmount(self,witam):
        witam = float(input("Enter amount to be taken out: "))
        if self.balance>=witam:
            self.balance -= witam
            print("\n Amount taken out:", witam)
            print("\n Amount left:", self.balance)
        else:
            print("\n Not much balance")
```

```
In [105... #creating customer class
class Customer_info(Bank_account):
    def __init__(self,acc_no,holder_name,acc_type,balance,address,mobile):
        self.acc_no=acc_no
        self.holder_name=holder_name
        self.acc_type=acc_type
        self.balance=balance
        self.address = address
        self.mobile = mobile
        # invoking the __init__ of the parent class
        Bank_account.__init__(self,acc_no,holder_name,acc_type,balance)

    def display(self):
        print('Account number:',self.acc_no)

        print('holder:',self.holder_name)

        print('Type:',self.acc_type)

        print('Balance:',self.balance)

        print('address:',self.address)

        print('mobile',self.mobile)
```

```
In [113... #creating branch class
class branch_info(Bank_account):
    def __init__(self,address=None,branch_no=None):
        self.address=address
        self.branch_no=branch_no

    def display(self):

        print('address:',self.address)
        print('branch_no',self.branch_no)
```

```
In [79]: #trial 1 base class
acc1=Bank_account('34567','IShan','C','500')
acc2=Bank_account('34568','ISha','S','5100000')

acc2.createAccount()
acc2.report()

Enter the account no : 34567
Enter the account holder name : IShan
Enter the type of account [C/S] : C
Enter The Initial amount(>=500 for Saving and >=1000 for current)500

Account Created
34567  IShan  C   500
```

```
In [56]: #seeing account
acc1.withdrawAmount(100)
acc1.depositAmount(2500000)

Enter amount to be taken out: 8800

Amount taken out: 8800.0

Amount left: 2491500.0
Enter amount to be deposited: 700000

Amount Deposited: 700000.0

Amount total: 3191500.0
```

```
In [106... #trial 2 inherited class cust detail
acc1= Customer_info('34567','IShan','C','500','kol',4467)

acc1.display()

Account number: 34567
holder: IShan
Type: C
Balance: 500
address: kol
mobile 4467
```