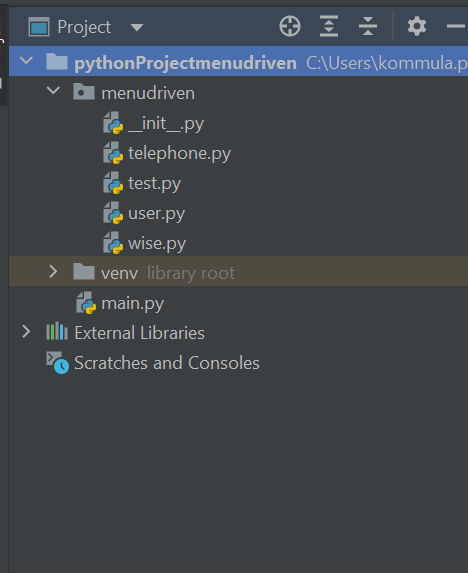
**MENU DRIVEN**

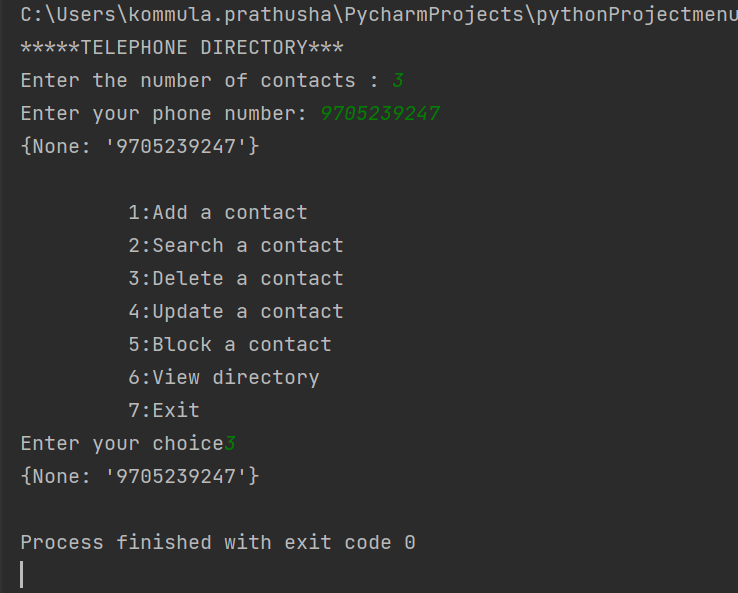


**Menudriven.py**

**Telephone.py**

def add(dict1):  
 name3= raw\_input("Enter the new name you want to add: ")  
 num3=input("Enter the number: ")  
 dict1[name3]=num3  
 print (dict1)  
  
  
def search(dict1,n,list1,temp):  
 name2= raw\_input("Enter the name whose number is to be found: ")  
 for i in range(0,name2):  
 if list1[i]==name2:  
 temp=i  
 if temp!=100:  
 print( "Number is : ", list2[temp])  
  
def delete(dict1):  
 name4= raw\_input("Enter the name you want to delete: ")  
 del dict1[name4]  
 print(dict1)  
  
def update(dict1,n,list1):  
 global temp  
 name5= raw\_input("Enter the name which you want to update: ")  
 for i in range(0,n):  
 if list1[i]==name5:  
 temp=i  
 if temp!=100:  
 num5=input("Enter the new number")  
 dict1[name5]=num5  
 print (dict1)  
def view(dict1):  
 print(dict1)  
print ("\*\*\*\*\*TELEPHONE DIRECTORY\*\*\*")  
list1=[]  
list2=[]  
dict1={}  
temp=100  
n=input("Enter the number of contacts : ")  
  
  
def raw\_input(param):  
 pass  
  
for i in range(1,2):  
 name1=raw\_input("Enter your name: ")  
 num=input("Enter your phone number: ")  
 list1.extend([name1])  
 list2.extend([num])  
 dict1=dict(zip(list1,list2))#to convert two list into dictionary  
print (dict1)  
  
print ("""  
 1:Add a contact  
 2:Search a contact  
 3:Delete a contact  
 4:Update a contact  
 5:View directory  
 6:Exit""")  
choice=input("Enter your choice")  
if choice == 1:  
 add(dict1)  
elif choice == 2:  
 search(dict1, n, list1, temp)  
elif choice == 3:  
 delete(dict1)  
elif choice == 4:  
 update(dict1, n, list1)  
else:  
 view(dict1)

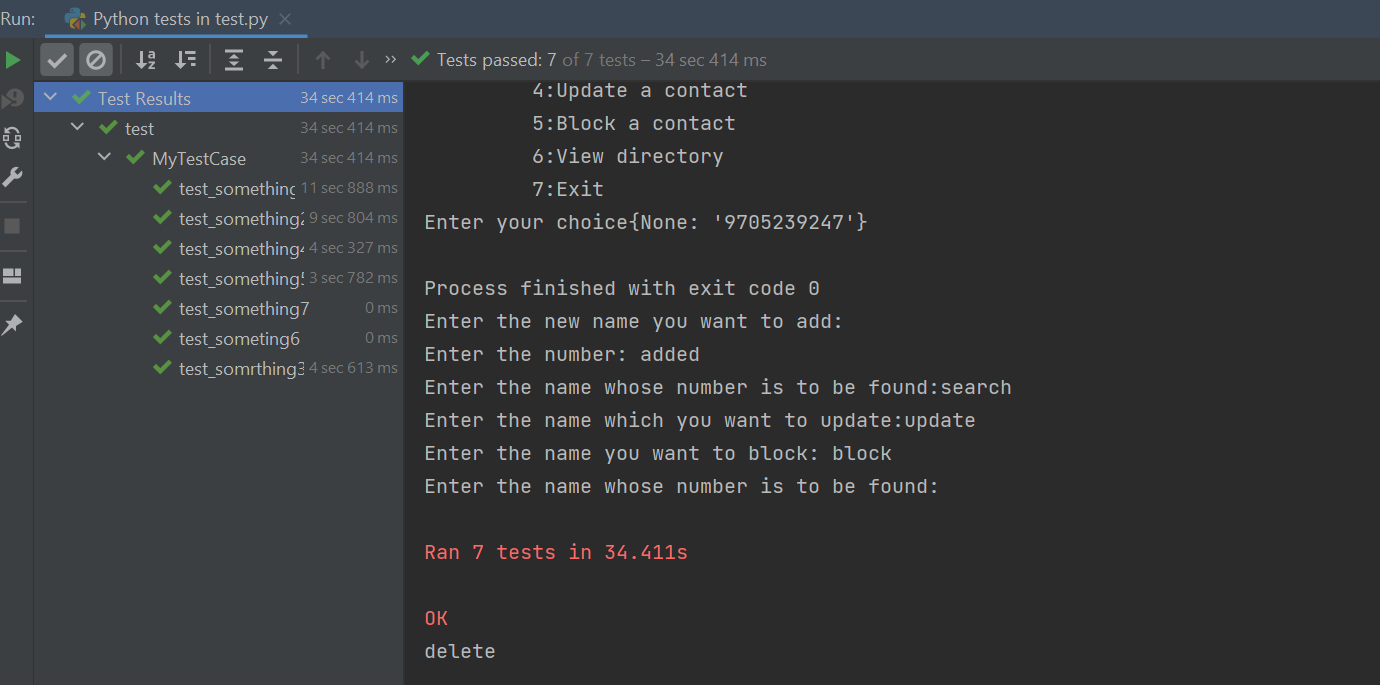
**output**

****

**test.py**

import unittest  
from menudriven import telephone  
  
  
  
class MyTestCase(unittest.TestCase):  
 def test\_something1(self):  
 exp1= input("Enter the new name you want to add: \n"),input("Enter the number: ")  
 act1 = print("added")  
 def test\_something2(self):  
 exp2= input("Enter the name whose number is to be found:")  
 act2= print("search")  
 def test\_somrthing3(self):  
 exp3= input("Enter the name whose number is to be found:")  
 act3 = print("delete")  
 def test\_something4(self):  
 exp4 = input("Enter the name which you want to update:")  
 act4= print("update")  
 def test\_something5(self):  
 exp5= "view"  
 act5= "view"  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 unittest.main()

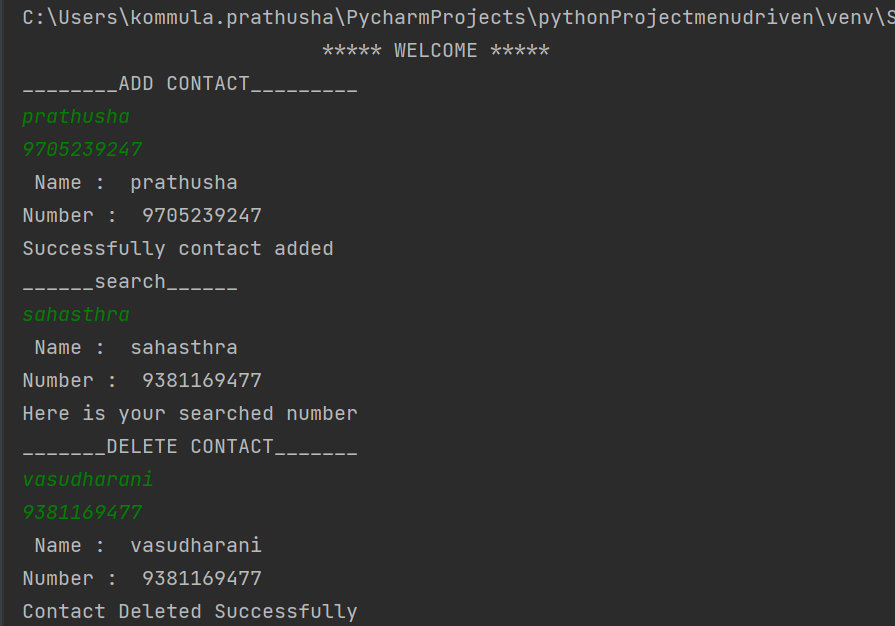
**output**

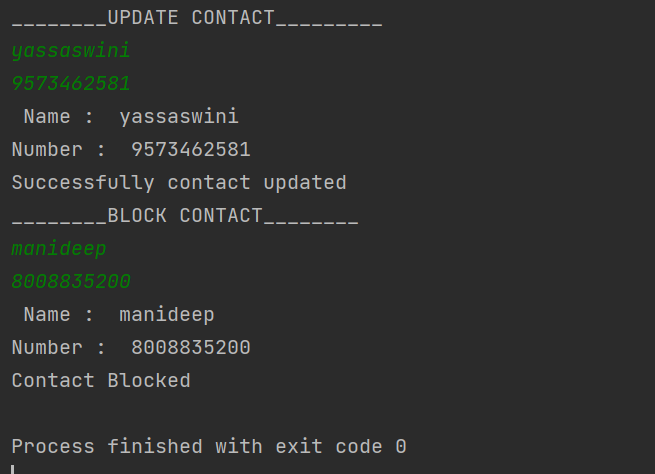
****

**user.py**

class person:  
 def set\_name(self, name):  
 self.name=name  
 def set\_id(self, id):  
 self.id=id  
 def get\_name(self):  
 return self.name  
 def get\_id(self):  
 return self.id  
print(" \*\*\*\*\* WELCOME \*\*\*\*\* ")  
  
print("\_\_\_\_\_\_\_\_ADD CONTACT\_\_\_\_\_\_\_\_\_")  
a=person()  
a.set\_name(input())  
a.set\_id(input())  
print(" Name : ", a.get\_name())  
print("Number : ", a.get\_id())  
print("Successfully contact added")  
  
print("\_\_\_\_\_\_search\_\_\_\_\_\_")  
b=person()  
b.set\_name(input())  
b.set\_id(9381169477)  
print(" Name : ", b.get\_name())  
print("Number : ", b.get\_id())  
print("Here is your searched number")  
  
print("\_\_\_\_\_\_\_DELETE CONTACT\_\_\_\_\_\_\_")  
c=person()  
c.set\_name(input())  
c.set\_id(input())  
print(" Name : ", c.get\_name())  
print("Number : ", c.get\_id())  
print("Contact Deleted Successfully")  
  
print("\_\_\_\_\_\_\_\_UPDATE CONTACT\_\_\_\_\_\_\_\_\_")  
d=person()  
d.set\_name(input())  
d.set\_id(input())  
print(" Name : ", d.get\_name())  
print("Number : ", d.get\_id())  
print("Successfully contact updated")  
  
print("\_\_\_\_\_\_\_\_BLOCK CONTACT\_\_\_\_\_\_\_\_")  
e=person()  
e.set\_name(input())  
e.set\_id(input())  
print(" Name : ", e.get\_name())  
print("Number : ", e.get\_id())  
print("Contact Blocked")

**output**

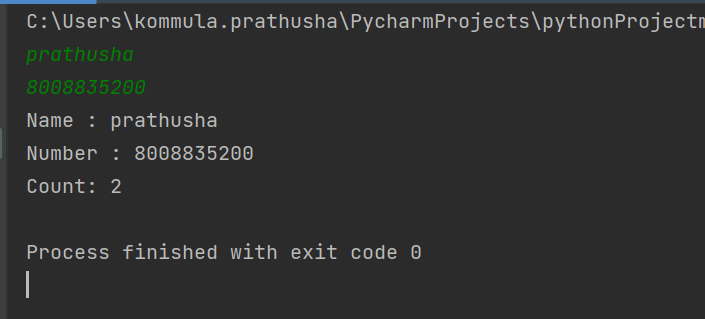
****

****

**wise.py**

class customer:  
 name = input() # Class attribute 1  
 number = input() # Class attribute 2  
 count = 0# Class atrribute 3  
  
 def show(self):  
 self.name  
 self.number  
 customer.count += 1  
  
  
c1 = customer()  
c1.show()  
c1.show()  
print("Name :", c1.name) # Accessing class attribute name  
print("Number :", c1.number) # Accessing class attributes age  
print("Count:", c1.count)

**output**

****