class Contact:

def \_\_init\_\_(self, name, phone, email):

self.name = name

self.phone = phone

self.email = email

def \_\_str\_\_(self):

return f" Name: {self.name}\n Phone: {self.phone}\n Email: {self.email}"

class ContactManager:

def \_\_init\_\_(self):

self.contacts = []

def add\_contact(self, contact):

self.contacts.append(contact)

def remove\_contact(self, name):

for contact in self.contacts:

if contact.name == name:

self.contacts.remove(contact)

print(f"{name} removed successfully.")

print(f"Contact {name} not found.")

def search\_contact(self, name):

for contact in self.contacts:

if contact.name == name:

return contact

return None

def display\_contacts(self):

if len(self.contacts) == 0:

print("No contacts found.")

else:

for contact in self.contacts:

print(contact)

Print("--------------------")

contact\_manager = ContactManager()

while True:

print("Contact Manager")

print("1) Add contact")

print("2) Remove contact")

print("3) Search contact")

print("4) Display contacts")

print("5) Exit")

choice = input("Enter your choice: ")

if choice == "1":

name = input("Enter name: ")

phone = input("Enter phone: ")

email = input("Enter email: ")

contact = Contact(name, phone, email)

contact\_manager.add\_contact(contact)

print("Contact added successfully.")

elif choice == "2":

name = input("Enter name: ")

contact\_manager.remove\_contact(name)

elif choice == "3":

name = input("Enter name: ")

contact = contact\_manager.search\_contact(name)

if contact is not None:

print(contact)

else:

print(f"Contact {name} not found.")

elif choice == "4":

contact\_manager.display\_contacts()

elif choice == "5":

break

else:

print("Wrong input!")