Lesson 8 Assignment

Laura Bartlett

**Source Code:**

#Lesson 8 Assignemnt

import csv

#function to create ncfile.csv for new contacts

def create\_f():

    with open("ncfile.csv","w") as f:

        writer = csv.writer(f)

        writer.writerow(["Name","Phone","Email"])

def new\_contact():

    contacts = []

    #collect user input for contact information

    name = input("What is the contact's name?  ")

    phone = input("What is the contact's phone number?  ")

    email = input("What is the contact's email address?  ")

    #create dictionary to store contact information

    contact\_info = {

        "name" : name,

        "phone" : phone,

        "email" : email

    }

    #add contact\_info in the contacts list

    contacts.append(contact\_info)

    #append new contact into ncfile.csv

    with open("ncfile.csv","a") as f:

        writer = csv.writer(f)

        for contact in contacts:

            writer.writerow([contact["name"], contact["phone"], contact["email"]])

#function to read and print contacts, using newline='' and [row for row in reader if any(row)] to eliminate the extra lines

#by only printing rows that are populated from the csv file. It took some research to get this to print without extra rows!

#I will appreciate feedback about how to keep the extra rows to be added into the CSV file if possible!

def print\_contacts():

    with open("ncfile.csv", "r", newline='') as f:

        reader = csv.reader(f)

        contact\_list = [row for row in reader if any(row)]

        for row in contact\_list:

            #format the output of the contacts file, used \t for spacing and element strip to get rid of the brackets that were printing

            formatted = "\t".join([element.strip() for element in row])

            print(formatted)

#funtion to edit rows in ncfile.csv

def edit():

    with open("ncfile.csv","r", newline='') as f:

        reader = csv.reader(f)

        #row for row if any(row) prints only non-blank rows from the file

        editable = [row for row in reader if any(row)]

    print("Current contacts list:")

    #index to show line numbers for editing, also format with elements.strip to take out the extra brackets, \t for spacing

    for idx, contact in enumerate(editable):

        formatted = "\t".join([element.strip() for element in contact])

        print(f"{idx} : {formatted}")

    #try statement to pick the row to edit

    try:

        edit\_row = int(input("Please enter the row number of the contact to edit:  "))

        if edit\_row < 0 or edit\_row >= len(editable):

            print("Invalid input. Try again.")

    except ValueError:

        print("Invalid Input")

    #get updated contact info for selected row

    name = input("Enter new name or press enter to skip: ")

    phone = input("Enter new phone number or press enter to skip: ")

    email = input("Enter new email address or press enter to skip: ")

    #if statements to determine which comma seperated value to assign changes to

    if name:

        editable[edit\_row][0] = name

    if phone:

        editable[edit\_row][1] = phone

    if email:

        editable[edit\_row][2] = email

    #write the changes to csv file

    with open("ncfile.csv","w") as f:

        writer = csv.writer(f)

        writer.writerows(editable)

    print("Congratulations! Your contact has been updated.")

#main function

def main():

    print("Welcome to Contact Manager\nKeep your contacts up to date and organized!\n")

    create\_f()

    new\_contact()

     #while loop to give the user options and run functions based on user's choice

    while True:

        print("\nContact Management Options: \n")

        print("1 - Make a new contact file")

        print("2 - Add a new contact")

        print("3 - View all contacts")

        print("4 - Edit an existing contact")

        print("5 - Save changes and exit Contact Manager")

        choice = input("\nPlease enter selection:  ")

        if choice == '1':

            create\_f()

        if choice == '2':

            new\_contact()

        if choice == '3':

            print\_contacts()

        if choice == '4':

            edit()

        if choice == '5':

            print("Thank you for using Contact Manager!")

            break

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

    main()

print("\nCompleted by Laura Bartlett\n")

**Screenshots:**



