**2-feedbacklista:**

**Antonio pardal López**

Este programa es prácticamente igual que el otro, creamos dos listas en vez de diccionarios.

clients = []  
vip = []

Las funciones en vez de importarlas y llamarlas las introducimos directamente en la función añadiendo bucles while true.

*def* crear\_cliente():  
 *#dni  
 while True*:  
 dni = input("introduce tu dni: ") *# definimos la variable que pedira datos al usuario* nif = dni[  
 :8] *# definimos una variable para contar dni de 0 hasta 8 que son los numeros que queremos contar para comprobar el dni* n = nif.isdigit() *# creamos variable que te dice si todos los strings de 0 a 8 son numericos* i = len(dni) *# variable para contanr el numero de caracteres  
 if* i != 9: *# si el numero de caracteres es diferente de 9* print("el dni debe contener 9 caracteres") *# imprimira esto  
 elif* n *is False*: *# o si n es falso* print("los 8 primeros caracteres deben ser numericos") *# imprimira esto  
 else*:  
 *break  
 #nombre* nombre = input("escribe tu nombre: ")  
 *#apellidos* apellidos = input("escribe tus apellidos: ")  
 *#direccion* direccion = input("escribe tu direccion: ")  
 *#telefono  
 while True*:  
 tlf = input("escribe tu tlf: ")  
 *if* len(tlf) < 9 *or* len(  
 tlf) > 9: *# si el numero de caracteres es menor o mayor que 9 entrara la siguiente linea sino entrara else:* print("el tlf debe contener 9 caracteres numericos")  
 *else*:  
 *break # sale de la iteracion  
 #email  
 while True*:  
 email = input("introduce tu direccion email: ")  
 *if* email.count("@hotmail") *or* email.count("@gmail") *or* email.count("@outlook") *and* email.count("."):  
 *break  
 else*:  
 print("el email es incorrecto")  
 *#fecha  
 while True*:  
 fecha = input("introduzca fecha en forma: 01/01/0001:")  
 i = len(fecha) *# creamos una variable para contar los caracteres* fechaSplit = fecha.split("/")  
 *if* i != 10: *# si los caracteres son diferentes de 10 imprimira la siguiente linea* print("la fecha debe contener 10 caracteres / incluidas")  
 *if* (*not* fecha.count("/")) *or* len(fechaSplit) != 3 *and* i >= 8 *and* i <= 10:  
 print("la fecha debe incluir minimo 2 /")  
 *else*:  
 *if* (fechaSplit[0].isdigit() *and* int(fechaSplit[0]) > 0 *and* int(fechaSplit[0]) < 32):  
 *if* (fechaSplit[1].isdigit() *and* int(fechaSplit[1]) > 0 *and* int(fechaSplit[1]) <= 12):  
 *if* (fechaSplit[2].isdigit() *and* int(fechaSplit[2]) < 2100 *and* int(fechaSplit[2]) >= 1920):  
 *break  
 else*:  
 print("año incorrecto")  
 *else*:  
 print("mes incorrecto")  
 *else*:  
 print("dia incorrecto")  
 *#tipo de cliente  
 while True*:  
 habitual = input("¿eres cliente habitual? (s/n): ")  
 *if* habitual == "s" *or* habitual == "S" *or* habitual == "SI" *or* habitual == "si":  
 habitual = *True #creamos el valor de la variable bool para poder comprobar el tipo de cliente mas adelante  
 break  
 elif* habitual == "n" *or* habitual == "N" *or* habitual == "NO" *or* habitual == "no":  
 habitual = *False  
 break  
 else*:  
 print("[Error] Ingrese si o no.")  
 *continue*

cambiamos de cliente habitual el tipo de cliente por un valor booleano. De esta manera comprobaremos a que lista deberá ir cada cliente y todos sus datos.

Cambiamos la forma de agregar los datos del cliente ya que ahora agregamos a listas y no a diccionarios. Con un if comprobaremos si habitual es True y si lo es, con el método extend()

Agregaremos los datos a la lista vip y a la de clients, añadimos a las dos listas para más adelante comprobar que en clientes aparezcan todos y vip para comprobar solo los clientes vip.

En caso contrario de que sea valor false solo agregara los datos a la lista de clients.

*#listas a las que iran los datos  
if* habitual:  
 vip.extend([[dni, nombre, apellidos, direccion, tlf, email, fecha, habitual]]) *#metodo extend para introducir varios parametros* clients.extend([[dni, nombre, apellidos, direccion, tlf, email, fecha, habitual]])  
*else*:  
 clients.extend([[dni, nombre, apellidos, direccion, tlf, email, fecha, habitual]])

**Opción 2: eliminar cliente:**

*def* eliminar\_cliente():  
 dni = input('Introduce dni del cliente: ')  
 *for* client *in* clients: *# dentro de clientes buscaremos un cliente  
 if* dni *in* client: *#si el dni esta en algun cliente  
 del* client[:] *#eliminara el cliente con ese dni y sus datos* print("el cliente con dni", dni, "se a eliminado")  
 *break  
 else*:  
 print('No existe el cliente con dni', dni)

con la variable DNI pedimos el DNI al usuario, comprobamos este con un bucle for para saber si dentro de clientes esta el cliente con ese DNI y si esta lo eliminamos con la función del, sino nos imprimirá un mensaje diciendo que el cliente no existe.

**Opción 3: mostrar cliente:**

*def* mostrar\_cliente():  
 dni = input('Introduce dni del cliente: ')  
 *for* client *in* clients:  
 *if* dni *in* client:  
 print(client[:]) *#mostrara todos los datos del cliente  
 break  
 else*:  
 print('No existe el cliente con ese dni', dni)

con la variable DNI pedimos el DNI al usuario, comprobamos este con un bucle for para saber si dentro de clientes está el cliente con ese DNI y si esta imprimirá todos los datos del cliente sino nos dirá que no existe.

**Opción 4: lista clientes:**

*def* lista\_clientes():  
 print('Lista de clientes')  
 *for* client *in* clients:  
 print(client[:2]) *#mostrara los dos primeros datos del cliente*

imprimiremos los dos primeros datos de cada cliente dentro de la lista clients.

**Opción 5: listas habituales:**

*def* lista\_habituales():  
 print('Lista de clientes habituales')  
 *for* v *in* vip:  
 print(v[:2])

imprimiremos los dos primeros datos de cada cliente dentro de la lista vip.