

EJERCICIOS CON DICCIONARIOS

1)

Escribir un programa que implemente una agenda. En la agenda se podrán guardar nombres y números de teléfono.

El programa nos dará el siguiente menú:

Añadir: Nos pide un nombre. Si el nombre se encuentra en la agenda, debe mostrar el teléfono. Si no, agregarlo.

Buscar: Nos pide una cadena de caracteres, y nos muestras todos los contactos cuyos nombres comiencen por dicha cadena.

Borrar: Nos pide un nombre y si existe nos preguntará si queremos borrarlo de la agenda.

Listar: Nos muestra todos los contactos de la agenda.

Para esto, tenemos que utilizar los métodos vistos en el apunte anterior y:

Crear un diccionario vacío que es el que permite almacenar los datos

Crear un menú para preguntarle al usuario que acción tomar

Si la acción es 1, le pedimos que ingrese un nombre y con el método `in` lo buscamos, si este devuelve verdadero mostramos el número de teléfono, si no, le pedimos que ingrese un número de teléfono y lo agregamos a nuestro diccionario.

Si la acción es 2, tenemos que usar el método `startswith` de una cadena, y si este devuelve verdadero mostramos el nombre.

Si la acción es 3, pedimos un nombre y con el método `in` lo buscamos, como en el primer punto, la diferencia es que si lo encuentra debemos borrarlo.

Si la acción es 4, mostramos todos los contactos del diccionario con su número, para esto podemos recorrer el diccionario normalmente como lo vimos en la última clase.

Si la acción es 0, debemos finalizar el programa, esto se puede hacer con la instrucción `break`.

Si la acción es diferente a las 5 anteriores, debemos mostrar que la acción no existe y permitirle ingresar otra.

2)

Permitirle a una rotisería ingresar un diccionario de comidas con sus precios hasta que ingrese

0, luego de eso deberemos preguntarle al usuario que comida desea comprar y con que cantidad, si la comida existe en el diccionario debemos devolverle el total que va a gastar y si la comida no existe debemos devolverle un mensaje de error.

Para esto, tenemos que hacer un `While` que permita ingresar elementos en un diccionario hasta que el nombre del elemento sea 0.

Después, si el elemento existe debemos multiplicar el precio de este por la cantidad ingresada por el usuario.