

Solucionario Ejercicios Propuestos - Funciones y Modularización

Solucionario Ejercicios Propuestos - Funciones y Modularización	1
Lectura - Condensando código	2
Ejercicio propuesto 1: Otra encuesta	2
Lectura - Anatomía de una función	3
Ejercicio propuesto 1: Transformación de unidad de medida de la Temperatura	3
Ejercicio propuesto 2: Recomendador	4
Lectura - Tipos de argumentos y parámetros	5
Ejercicio Propuesto 1: Normalización	5
Lectura - Alcance de variables	6
Ejercicio propuesto 1: La revancha	6
Lectura - Modularización	7
Ejercicio Propuesto 1: Menú Cajero Automático	7

Lectura - Condensando código

Ejercicio propuesto 1: Otra encuesta

```
def imprimir_menu_3():
    print('Opciones: ')
    print('1). De acuerdo')
    print('2). En desacuerdo')
    print('3). No me interesa')
def imprimir_menu_4():
    print('Opciones: ')
    print('1). De acuerdo')
    print('2). En desacuerdo')
    print('3). No me interesa')
    print('4). Prefiero no Responder')

preguntas = ['Enunciado Pregunta 1',
             'Enunciado Pregunta 2',
             'Enunciado Pregunta 3',
             'Enunciado Pregunta 4',
             'Enunciado Pregunta 5',
             'Enunciado Pregunta 6',
             'Enunciado Pregunta 7',
             'Enunciado Pregunta 8',
             'Enunciado Pregunta 9',
             'Enunciado Pregunta 10']
respuestas = []

# Hacer preguntas
for i, p in enumerate(preguntas):
    print(p)
    if i in [2,5,8]:
        imprimir_menu_4()
    else:
        imprimir_menu_3()
    respuestas.append(input('> '))

# Mostrar respuestas
for k in range(9):
    print(f'La respuesta a la pregunta {k+1} fue {respuestas[k]}')

print('Muchas gracias por responder la encuesta')
```

Lectura - Anatomía de una función

Ejercicio propuesto 1: Transformación de unidad de medida de la Temperatura

0° Fahrenheit

```
def to_celcius(f):  
    return (f-32)/1.8  
  
print(to_celcius(0))
```

Ebullición en Fahrenheit

```
def to_fahrenheit(c):  
    return 1.8*c + 32  
  
to_fahrenheit(100)
```

Ejercicio propuesto 2: Recomendador

```
import random

def elegir_tv(eleccion):
    peliculas = ['Scarface', 'Crepúsculo', 'Up']
    series = ['Dark', 'Friends', 'El Reemplazante']

    if eleccion == '1':
        print(f'Te recomiendo ver {random.choice(peliculas)}')
    elif eleccion == '2':
        print(f'Te recomiendo ver {random.choice(series)}')
    else:
        print(f'Ingresaste una opción inválida. No puedo recomendar nada.')

def elegir_cocina(eleccion):
    comidas = ['Filete Mignon', 'Cazuela', 'Bistec a lo Pobre']
    postres = ['Creme Brulee', 'Macedonia', 'Tiramisú']

    if eleccion == '1':
        print(f'Te recomiendo cocinar {random.choice(comidas)}')
    elif eleccion == '2':
        print(f'Te recomiendo cocinar {random.choice(postres)}')
    else:
        print(f'Ingresaste una opción inválida. No puedo recomendar nada.')

def submenu(eleccion):
    if eleccion == '1':
        texto = '''Elige entre:
        1. Comidas
        2. Postres
        > '''
    else:
        texto = '''Elige entre:
        1. Películas
        2. Series
        > '''

    return input(texto)
```

```
opcion = input('¿Estás aburrido? ¿Qué quieres hacer?')

    1. Cocinar
    2. Ver Televisión
    > '')

opcion2 = submenu(opcion)

if opcion == '1':
    elegir_cocina(opcion2)
elif opcion == '2':
    elegir_tv(opcion2)
```

Lectura - Tipos de argumentos y parámetros

Ejercicio Propuesto 1: Normalización

```
import math
def modulo(lista):
    return math.sqrt(sum([elemento**2 for elemento in lista]))

def normalizar(lista):
    mod = modulo(lista)
    return [elemento/mod for elemento in lista]

vector = [1, 2, 3, 4, 5, 6]
print(normalizar(vector))
```

Lectura - Alcance de variables

Ejercicio propuesto 1: La revancha

```
import random

# pool de números
pool_loto = [n for n in range(1,42)]
pool_revancha = [n for n in range(1,42)]

# la posición de extracción
posiciones = ['primer número',
              'segundo número',
              'tercer número',
              'cuarto número',
              'quinto número',
              'sexto número',
              'comodín']

def sacar_numero(posicion, sorteo):

    if sorteo == 'Loto':
        global pool_loto
        elegido = random.choice(pool_loto)
        pool_loto.remove(elegido)
    else:
        global pool_revancha
        elegido = random.choice(pool_revancha)
        pool_revancha.remove(elegido)

    print(f'{sorteo}: El {posicion} es {elegido}')

def sortear(sorteo, posiciones = posiciones):
    if sorteo == 'Revancha':
        posiciones = posiciones[:-1]

    for pos in posiciones:
        sacar_numero(pos, sorteo)

eleccion = input('Bienvenidos. ¿Qué desea jugar?')
```

```
1. Loto
2. Loto + Revancha
> '')

# Elección de Sorteos
if eleccion == '1':
    sortear('Loto')
elif eleccion == '2':
    sortear('Loto')
    sortear('Revancha')
else:
    print('Elección no válida')

print('Sorteo Terminado')
```

Lectura - Modularización

Ejercicio Propuesto 1: Menú Cajero Automático

Módulo `deposito.py`

```
def depositar(saldo, cantidad):
    total = saldo + cantidad
    print(f'Su nuevo saldo es: ${total}')
    return total

if __name__ == '__main__':
    saldo = 10000
    cantidad = 1000
    depositar(saldo, cantidad)
```

Módulo giro.py

```
def girar(saldo, cantidad):  
    total = saldo - cantidad  
    print(f'Su nuevo saldo es: ${total}')  
    return total  
  
if __name__ == '__main__':  
    saldo = 10000  
    cantidad = 1000  
    depositar(saldo, cantidad)
```

Módulo chequeo.py

```
from giro import girar  
  
def check(saldo, cantidad, funcion):  
    if saldo < cantidad:  
        print(f'No se puede aplicar la operación solicitada. Su saldo es  
            ${saldo}')        error = True  
    else:  
        saldo = funcion(saldo, cantidad)  
        error = False  
  
    return saldo, error  
  
if __name__ == '__main__':  
    saldo = 10000  
    cantidad1 = 11000  
    saldo, error = check(saldo, cantidad1, girar)  
    cantidad2 = 9000  
    saldo, error = check(saldo, cantidad2, girar)
```


main.py

```
import sys

from deposito import depositar
from giro import girar
from chequeo import check

menu_principal = '''¡Bienvenido al Banco Amigo!. Escoja una opción:
    1. Consultar saldo
    2. Hacer depósito
    3. Realizar giro
    4. Salir
> '''

def mostrar_menu(saldo = 0):
    opcion = ''
    while opcion != 4:

        opcion = int(input(menu_principal))

        if opcion > 0 and opcion < 5:
            if opcion == 1:
                print(f"Su saldo es de: {saldo}")

            elif opcion == 2:
                cantidad = int(input("Ingrese cantidad: "))
                saldo = depositar(saldo, cantidad)

            elif opcion == 3:
                if saldo > 0:
                    error = True
                    while error:
                        cantidad = int(input("Ingrese cantidad: "))
                        saldo, error = check(saldo, cantidad, girar)
                else:
                    print("No puede realizar giros. Su saldo es de 0.")

            elif opcion == 4:
                print("Saliendo")

        else:
            print("Opción inválida. Ingrese 1, 2, 3 ó 4.")
```

```
if len(sys.argv) == 2:  
    mostrar_menu(int(sys.argv[1]))  
else:  
    mostrar_menu()
```