

Ejercicios de Programación

Ejercicio 1: Aplicar Descuento

Este ejercicio tiene como objetivo ayudarte a comprender cómo calcular el total de una cuenta después de aplicar un descuento. Es ideal para practicar operaciones matemáticas simples y trabajar con funciones.

Instrucciones:

1. Define una función que tome como parámetros el total de una cuenta y un porcentaje de descuento.
2. Calcula el monto correspondiente al descuento.
3. Resta el descuento del total original.
4. Devuelve el total final.

Casos de ejemplo:

- Si tienes una cuenta de \$1000 y un descuento del 20%, el resultado debe ser \$800.
- ¿Qué pasa si aplicas un descuento del 0%? ¿Y del 100%?

Solución propuesta:

```
def aplicar_descuento(total_cuenta, porcentaje_descuento):  
    descuento = total_cuenta * (porcentaje_descuento / 100)  
    total_con_descuento = total_cuenta - descuento  
    return total_con_descuento  
  
# Ejemplo de uso  
total = 1000  
descuento = 20  
total_final = aplicar_descuento(total, descuento)  
print(f"El total después del descuento es: {total_final}")
```

Ejercicio 2: Conversor de Unidades de Tiempo

Este ejercicio te permitirá practicar cómo convertir valores entre diferentes unidades de tiempo. Es una excelente oportunidad para trabajar con estructuras condicionales y comprender las relaciones entre unidades de tiempo.

Instrucciones:

1. Define tres funciones:
 - `a_segundos`: Convierte una cantidad y unidad dada a segundos.
 - `de_segundos`: Convierte una cantidad en segundos a otra unidad deseada.
 - `convertir_tiempo`: Usa las dos funciones anteriores para realizar una conversión completa entre dos unidades.
2. Considera las siguientes unidades:
3. segundo, minuto, hora, día, mes (30 días), año (365 días).

Casos de ejemplo:

- Convierte 2 horas a minutos: ¿Cuántos minutos son?
- Intenta convertir 5 lustros a años: ¿Qué mensaje de error obtienes?

Solución propuesta:

```
def a_segundos(cantidad, unidad):
    if unidad == 'segundo':
        return cantidad
    elif unidad == 'minuto':
        return cantidad * 60
    elif unidad == 'hora':
        return cantidad * 3600
    elif unidad == 'dia':
        return cantidad * 86400
    elif unidad == 'mes':
        return cantidad * 2592000 # Aproximadamente 30 días
    elif unidad == 'año':
        return cantidad * 31536000 # Aproximadamente 365 días
    else:
        return "Error"

def de_segundos(cantidad, unidad):
    if unidad == 'segundo':
        return cantidad
    elif unidad == 'minuto':
        return cantidad / 60
    elif unidad == 'hora':
        return cantidad / 3600
    elif unidad == 'dia':
        return cantidad / 86400
    elif unidad == 'mes':
        return cantidad / 2592000 # Aproximadamente 30 días
    elif unidad == 'año':
        return cantidad / 31536000 # Aproximadamente 365 días
    else:
        return "Error"

def convertir_tiempo(cantidad, unidad_origen, unidad_destino):
    cantidad_en_segundos = a_segundos(cantidad, unidad_origen)
    if cantidad_en_segundos == "Error":
        return "Unidad de tiempo no válida"
    cantidad_convertida = de_segundos(cantidad_en_segundos, unidad_destino)
    return f"{cantidad} {unidad_origen}(s) son {cantidad_convertida} {unidad_destino}(s)"

# Ejemplo de uso
print(convertir_tiempo(2, 'hora', 'minuto'))
```

Ejercicio 3: Calcular Promedio

En este ejercicio aprenderás a calcular el promedio de una serie de números. También trabajarás con parámetros variables en funciones.

Instrucciones:

1. Define una función que acepte un número variable de argumentos.
2. Si no se proporcionan números, devuelve 0.
3. Si se proporcionan números, calcula y devuelve el promedio.

Casos de ejemplo:

- Calcula el promedio de los números 1, 2, 3, 4, 5.
- ¿Qué sucede si no proporcionas ningún número?

Solución propuesta:

```
def calcular_media(*numeros):  
    if not numeros:  
        return 0  
    return sum(numeros) / len(numeros)  
  
# Ejemplo de uso  
print(calcular_media(1, 2, 3, 4, 5)) # Salida: 3.0
```

Ejercicio 4: Tirar Dados

Este ejercicio simula el lanzamiento de un dado y calcula la frecuencia con la que aparece cada cara. Es perfecto para trabajar con estructuras de control y bucles.

El siguiente código utiliza un módulo del lenguaje python para generar un número aleatorio entre 1 y 6, este lo veremos en mayor profundidad más adelante:

```
import random # esto va en la primera línea del archivo  
  
resultado = random.randint(1, 6)  
  
print(resultado)
```

Puedes utilizar esta misma línea de código en una función de esta manera:

```
def dameNumero(desde, hasta):  
    return random.randint(desde, hasta)  
  
print(dameNumero(1, 6))
```

Instrucciones:

Utilizando el código de más arriba,

1. Define una función que simule lanzar un dado múltiples veces.
2. Cuenta cuántas veces aparece cada cara del dado.
3. Calcula el porcentaje de ocurrencia de cada cara.
4. Maneja errores si el número de lanzamientos no es válido (por ejemplo, 0 o números negativos).

Casos de ejemplo:

- Lanza un dado 10 veces. ¿Qué porcentajes obtienes?
- Intenta lanzar un dado 0 veces. ¿Qué mensaje de error aparece?
- ¿Qué sucede si lanzas un dado 10000 veces? ¿Son los porcentajes cercanos al 16.67%?
- ¿Qué ocurre cuando lanzas el dado 1 vez?

Soluciones propuestas:**Solución 1**

```
import random

def tirar_dados(veces):
    cara_1 = 0
    cara_2 = 0
    cara_3 = 0
    cara_4 = 0
    cara_5 = 0
    cara_6 = 0

    if veces == 1:
        resultado = random.randint(1, 6)
        return print(f"Salió la cara: {resultado}")

    if veces <= 0:
        return print("Error: El número de lanzamientos debe ser mayor a 0.")

    for _ in range(veces):
        resultado = random.randint(1, 6)
        if resultado == 1:
            cara_1 += 1
        elif resultado == 2:
            cara_2 += 1
        elif resultado == 3:
            cara_3 += 1
        elif resultado == 4:
            cara_4 += 1
        elif resultado == 5:
            cara_5 += 1
        elif resultado == 6:
            cara_6 += 1

    cara_1 = (cara_1 / veces) * 100
    cara_2 = (cara_2 / veces) * 100
    cara_3 = (cara_3 / veces) * 100
    cara_4 = (cara_4 / veces) * 100
    cara_5 = (cara_5 / veces) * 100
    cara_6 = (cara_6 / veces) * 100

    print(f"Porcentaje de veces que salió la cara 1: {cara_1:.2f}%")
    print(f"Porcentaje de veces que salió la cara 2: {cara_2:.2f}%")
    print(f"Porcentaje de veces que salió la cara 3: {cara_3:.2f}%")
    print(f"Porcentaje de veces que salió la cara 4: {cara_4:.2f}%")
    print(f"Porcentaje de veces que salió la cara 5: {cara_5:.2f}%")
    print(f"Porcentaje de veces que salió la cara 6: {cara_6:.2f}%")
```

Solución 2

```
import random

def tirar_dados(veces):
    frecuencias = {i: 0 for i in range(1, 7)}
    if veces <= 0:
        return "Error: El número de lanzamientos debe ser mayor a 0."
    for _ in range(veces):
        resultado = random.randint(1, 6)
        frecuencias[resultado] += 1
    for cara, frecuencia in frecuencias.items():
        porcentaje = (frecuencia / veces) * 100
        print(f"Porcentaje de veces que salió la cara {cara}: {porcentaje:.2f}%")

# Ejemplo de uso
tirar_dados(10)
```