

Funciones

Las funciones son bloques de código que realizan tareas específicas y pueden ser reutilizadas en diferentes partes de un programa. Las funciones son importantes porque ayudan a organizar y estructurar el código, lo que facilita su mantenimiento y evolución. Además, el uso de funciones permite crear código más legible, eficiente y modular.

Los parámetros son valores que se pasan a una función para que pueda realizar su tarea. Los parámetros pueden ser de diferentes tipos, como enteros, flotantes, cadenas, listas, diccionarios, entre otros. Los parámetros permiten a las funciones ser más versátiles y personalizables para diferentes situaciones.

La devolución de un valor se refiere a la capacidad de una función para devolver un resultado específico después de realizar una tarea. El valor de retorno puede ser cualquier tipo de dato, como un número, una cadena, una lista, un diccionario, entre otros. La devolución de un valor es importante porque permite a una función comunicar su resultado a otras partes del programa.

Las buenas prácticas para definir funciones incluyen darles nombres descriptivos y significativos, utilizar verbos para indicar su acción, seguir las reglas de indentación y separar los nombres con guiones bajos en lugar de espacios. Además, es importante documentar las funciones con docstrings que expliquen su uso y comportamiento.

Aquí va un ejemplo de cómo se define una función en Python:

```
def calcular_promedio(lista):  
    """  
    Calcula el promedio de una lista de números.  
  
    Parámetros:  
    lista (list): Una lista de números.  
  
    Devuelve:  
    float: El promedio de la lista.  
    """  
    suma = sum(lista)  
    promedio = suma / len(lista)  
    return promedio
```

GUIA EJERCICIOS BASICOS FUNCIONES

1. Crear una función que convierta grados Celsius a grados Fahrenheit. Recibe un parámetro (grados Celsius) y devuelve el resultado en grados Fahrenheit.
2. Crear una función que calcule el área de un círculo. Recibe un parámetro (radio) y devuelve el área del círculo.
3. Crear una función que calcule el descuento aplicado a un producto. Recibe dos parámetros (precio original y precio con descuento) y devuelve el porcentaje de descuento aplicado.
4. Crear una función que calcule el promedio de edad de un grupo de personas. Recibe una lista de edades y devuelve el promedio.

5. Crear una función que determine si un número es primo o no. Recibe un parámetro (número) y devuelve True si es primo y False si no lo es.
6. Crear una función que calcule el área de un triángulo. Recibe dos parámetros (base y altura) y devuelve el área.
7. Crear una función que calcule el máximo común divisor de dos números. Recibe dos parámetros (números) y devuelve el máximo común divisor.
8. Crear una función que verifique si un número es par o impar. Recibe un número como parámetro y devuelve True si es par o False si es impar.
9. Crear una función que cuente la cantidad de veces que aparece un elemento en una lista. Recibe una lista y un elemento como parámetros y devuelve la cantidad de veces que aparece en la lista.
10. Crea una función que reciba como parámetros una lista de palabras y devuelva la palabra más larga.
11. Crea una función que reciba como parámetro una cadena de texto y devuelva la cantidad de vocales que tiene.
12. Crea una función que reciba dos listas de nombres, y devuelva una lista con los nombres que aparecen en ambas listas
13. Crear una función que recibe una lista de palabras y devuelve un diccionario con las palabras como llaves y su longitud como valores.

14. Crear una función que recibe una lista de números y devuelve un diccionario con el valor mínimo, el valor máximo y el promedio de los números en la lista.
15. Crear una función que recibe una lista de diccionarios con información de películas (título, género, director) y devuelve un diccionario con la cantidad de películas por género.