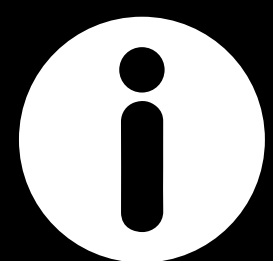


decoradores

Los decoradores son patrones de diseño en Python utilizados para dar nueva funcionalidad a objetos (funciones), modificando su comportamiento sin alterar su estructura: son funciones que modifican funciones.



Las funciones en Python soportan operaciones tales como ser asignadas a una variable, pasadas como argumento, y ser devueltas por otra función como resultado. También, es posible definir funciones dentro de funciones, sin que estén disponibles fuera de la función dentro de la cual fueron definidas. Los decoradores permiten que una función se modifique ante determinados escenarios, sin duplicar código.

```
def mostrar_informacion(function):  
  
    def interior():  
        print(f'Ejecutando la función {function.__name__}')  
        function()  
        print('Ejecución finalizada')  
  
    return interior  
  
def impresion():  
  
    print("Hola Mundo")  
  
funcion_decorada = mostrar_informacion(impresion)  
  
funcion_decorada()  
  
>>Ejecutando la función impresion  
>>Hola Mundo  
>>Ejecución finalizada
```

> Función como parámetro de una función

> Definición de una función dentro de otra

> Función que devuelve otra función como resultado

> Se asigna una función a una variable

> Ejecución de función decorada

> Funcionalidad extendida