

L U M B R E

```
function decorate(event) {  
  event = event || window.event;  
  var target = event.target || event.srcElement;  
  if (target && (target.getAttribute('action') || t  
    ga(function (tracker) {  
      var linkerParam = tracker.get('linkerParam');  
      document.cookie = '_shopify_ga=' + linkerParam  
    });  
}
```

Funciones y retorno de valores

Grado: 1

Materia: Fundamentos de Programación · Programación

11/10/2026

Antes de empezar

Una función es un nombre para un trozo de lógica. Encapsular cálculos en funciones con entradas y salidas claras hace el código reutilizable, testeable y legible, y es la base sobre la que se construyen programas que no se pudren.

QUÉ APRENDERÁS

- Definir funciones con parámetros y valores de retorno, usar argumentos por defecto y nombrados, y estructurar programas en unidades reutilizables.

Funciones y retorno de valores

Dar nombre a lo que se repite

Si copias y pegas un bloque de código más de una vez, debería ser una función. Las funciones son el mecanismo por el que un programa deja de ser una lista de instrucciones y pasa a ser un vocabulario de acciones con nombre.

Una función agrupa instrucciones bajo un nombre, recibe valores de entrada (parámetros) y devuelve un resultado. Se define con `def`, un nombre, una lista de parámetros entre paréntesis y un cuerpo indentado. Se usa llamándola con argumentos reales.

VIDEO

La sentencia `return` en Python

Qué devuelve una función y por qué imprimir no es lo mismo que retornar.



Escanea para verlo en Lumbre

<https://www.youtube.com/watch?v=ZnBQfF5JFDM>

Parámetros y argumentos

- El parámetro es el nombre en la definición; el argumento, el valor que se pasa al llamar.
- Se pueden pasar por posición o por nombre para claridad.
- Un valor por defecto hace opcional un parámetro.

El return y su ausencia

return entrega un resultado y termina la función de inmediato. Sin ningún return, la función devuelve el valor nulo None. Distinguir entre imprimir y devolver es crucial: solo lo devuelto puede seguir procesándose.

Devolver varios valores

Python permite devolver una tupla de varios resultados, que se recogen desempaquetando en varias variables. Es cómodo para funciones que calculan a la vez, por ejemplo, media y desviación.

Documentar con docstring

- La cadena inicial describe qué hace la función y qué espera.
- Sirve como ayuda interactiva y como contrato para quien la llama.
- Un buen docstring ahorra leer el cuerpo entero.

Funciones puras y sus ventajas

Una función pura devuelve siempre el mismo resultado para la misma entrada y no tiene efectos secundarios. Se prueba fácil, se paraleliza sin miedo y no esconde sorpresas; conviene preferirla salvo que el efecto sea necesario.

Descomponer problemas

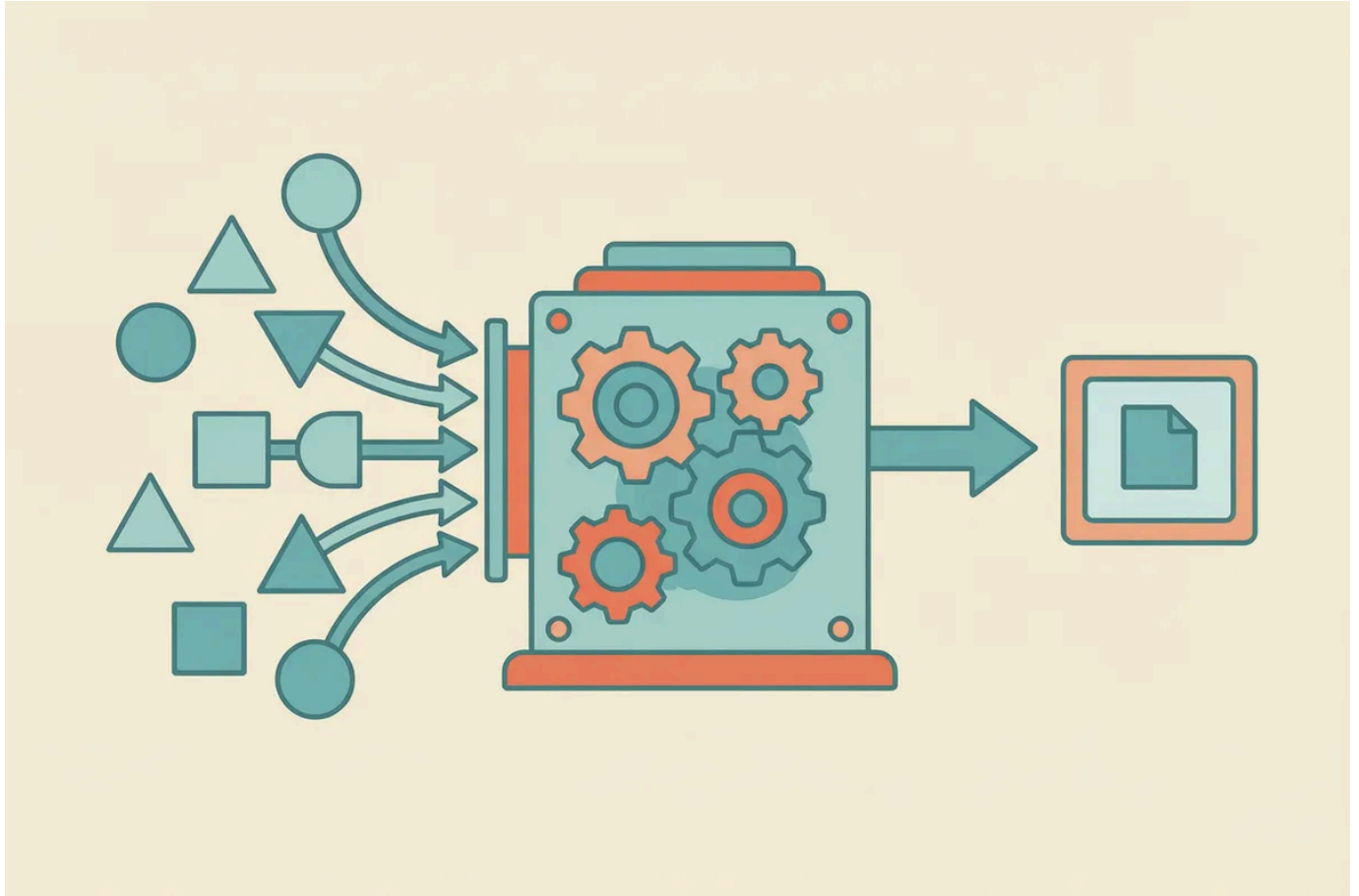
- Divide y vencerás: subproblemas pequeños, funciones independientes.
- Cada función hace una sola cosa y la hace bien.
- Los nombres de funciones leídas de seguido cuentan la historia del programa.

Abstracción y ocultamiento

Quien usa una función no necesita saber cómo la implementa, solo qué le da y qué le devuelve. Esa pared entre uso e implementación es lo que permite cambiar los detalles sin romper nada.

El paso de argumentos y sus sorpresas

Los objetos mutables pasados como argumentos pueden modificarse dentro y afectar fuera. Esa compartición es potente pero fuente de bugs: una función que altera la lista que recibió cambia el dato del llamante.



Una función recibe argumentos, procesa y devuelve un valor con return para poder seguir usándolo.

Reutilización en proyectos de datos

- Encapsular una transformación para aplicarla a varias columnas.
- Una función de carga centraliza la lectura y el saneado.
- Definir métricas como funciones hace comparable cualquier experimento.

Errores típicos

- Confundir el valor impreso en pantalla con el valor devuelto.
- Olvidar return y recibir None donde se esperaba un número.
- Cambiar una lista recibida creyendo que era una copia local.

Ejemplo resuelto: una función para normalizar columnas

Repetimos el mismo cálculo de restar la media y dividir por la desviación en varias columnas. Conviene encapsularlo en una función.

1. Definir la función con `def` y un parámetro que será la columna de entrada.
2. Calcular dentro la media y la desviación de esa columna.
3. Devolver la columna transformada con `return`, no imprimirla.
4. Llamar la función pasándole cada columna y guardar el resultado.
5. Comprobar que devolver permite encadenar otras operaciones sobre el valor obtenido.

¿Para qué sirve en la realidad?

Encapsular una transformación como función evita duplicar código, permite probarla por separado y la hace reutilizable sobre cualquier columna o experimento, que es la base de un pipeline mantenible.

1. Si una función no ejecuta ninguna sentencia `return`, su valor devuelto es:

- ☐ cero
- ☐ None
- ☐ el último cálculo hecho
- ☐ un error

2. Imprimir un valor dentro de una función equivale a devolverlo.

- ☐ Verdadero
- ☐ Falso

ACTIVIDAD INTERACTIVA DISPONIBLE EN LUMBRE

3. Funciones



4. Ordena los pasos para definir y usar una función correctamente:

Elementos:

- escribir def con el nombre y el cuerpo
- decidir qué parámetros recibe
- hacer el trabajo dentro del cuerpo
- entregar el resultado con return
- invocar la función con argumentos

Definir	. . .
Parametrizar	. . .
Calcular	. . .
Devolver	. . .
Lllamar	. . .

Subrutina <https://en.wikipedia.org/wiki/Subroutine> Secuencia de instrucciones empaquetada bajo un nombre reutilizable.

5. Une cada término con su papel en una función:

5.	el nombre en la definición	-----
5.	el valor que se pasa al llamar	-----
5.	entrega el resultado y termina	-----
5.	valor devuelto si no hay return	-----

Opciones: parámetro · argumento · return · None

Funciones y retorno

Funciones

Definición

def, nombre y parámetros

docstring

Retorno

return entrega y termina

sin return devuelve None

imprimir no es devolver

Diseño

una cosa bien hecha

funciones puras

abstracción

El mapa mental interactivo es interactivo en la versión online.



Función (programación) [https://en.wikipedia.org/wiki/Function_\(computer_programming\)](https://en.wikipedia.org/wiki/Function_(computer_programming))

Bloque de código reutilizable que recibe argumentos y devuelve un resultado.

6. Tienes tres bloques de código que normalizan columnas de forma parecida. Propón cómo convertirlos en una sola función reutilizable: qué parámetros recibiría y qué devolvería.

Referencia curricular

Grado:	1
Materia:	Fundamentos de Programación · Programación

Versión interactiva completa (con actividades, videos y autoevaluación):



<https://univ.lumbre.lat/object/cdia-prog1-008-funciones-y-retorno-de-valores>

Material educativo generado desde Lumbre · lumbre.lat