

Tipos en Parámetros

Es una buena práctica especificar los tipos de datos de los parámetros de las funciones en Python, especialmente en código que se espera que sea mantenido a lo largo del tiempo o que pueda ser utilizado por otros desarrolladores.

Ejemplo:

```
def suma(a: int, b: int) -> int:
    """ Esta función toma dos enteros como entrada y devuelve su suma. """
    return a + b
```

```
# Ejemplo de uso de la función
resultado = suma(3, 5)
print("El resultado de la suma es:", resultado)
```

Aunque Python es un lenguaje de programación dinámico y no requiere explícitamente que se especifiquen los tipos de datos, agregar anotaciones de tipos (disponibles desde Python 3.5 en adelante) puede ser muy útil, especialmente en proyectos más grandes o colaborativos.

Comentarios explicativos

Además, podemos utilizar los [comentarios](#) de varias líneas para explicar el propósito de una función, incluyendo la descripción de los parámetros de entrada y el valor de retorno.

Ejemplo:

```
def calculate_area(radius: float) -> float:
    """
    Calculates the area of a circle given its radius.

    Args:
        radius (float): The radius of the circle.

    Returns:
        float: The area of the circle.
    """
    # Formula for area of a circle:  $\pi * radius^2$ 
    area = 3.14159 * radius ** 2
    return area
```

```
# Example usage of the function  
radius = 5.0  
area = calculate_area(radius)  
print(f"The area of the circle with radius {radius} is {area}.")
```

Los [comentarios](#) están escritos en inglés para mantener consistencia y facilitar la colaboración con otros desarrolladores (recomendado).