

La importancia de los comentarios y los docstrings

¿Qué son los comentarios?

Los **comentarios** son líneas de texto dentro del código que no se ejecutan. Se utilizan para **explicar qué hace el código, por qué se tomó cierta decisión o cómo se debe usar una función o clase**.

En Python, se escriben precedidos por el símbolo #.

Ejemplo:

```
# Calcula el área de un círculo a partir del radio
```

```
area = 3.14 * radio ** 2
```

¿Por qué son importantes los comentarios?

1. **Mejoran la legibilidad:** permiten que otras personas (o el mismo programador tiempo después) entiendan el propósito del código.
2. **Facilitan el mantenimiento:** si hay que corregir o modificar algo, los comentarios orientan sobre cómo está estructurado el programa.
3. **Documentan decisiones de diseño:** explican por qué se eligió una determinada lógica o estructura.
4. **Aceleran el trabajo en equipo:** los desarrolladores pueden comprender rápidamente partes del código sin tener que analizar cada línea.
5. **Favorecen el aprendizaje:** los estudiantes y principiantes pueden comprender los ejemplos con más facilidad.

Buenas prácticas al comentar

- Comentá el **“por qué”** más que el “qué”: el código ya muestra *qué hace*, pero no siempre *por qué lo hace así*.
- Mantené los comentarios **actualizados**. Un comentario incorrecto es peor que ninguno.
- No comentes en exceso: demasiados comentarios pueden volver el código confuso.
- Usá un **tono claro y conciso**.

Ejemplo de buen comentario:

```
# Usamos una lista en lugar de un conjunto porque el orden de inserción es importante
```

¿Qué son los docstrings?

Un **docstring** (document string) es una **cadena de documentación** que describe el propósito y uso de un módulo, clase, método o función.

Se escribe entre comillas triples (""" """) y puede abarcar varias líneas.

Ejemplo:

```
def calcular_area_circulo(radio):
    """
    Calcula el área de un círculo dado su radio.

    Parámetros:
        radio (float): radio del círculo.

    Retorna:
        float: área del círculo.
    """
    return 3.1416 * radio ** 2
```

Ventajas de los docstrings



- Sirven como **documentación oficial** del código.
- Pueden ser accedidos con la función `help()` en Python.
- Herramientas como **Sphinx** o **pydoc** los utilizan para generar documentación automática.
- Facilitan la comprensión del propósito y los parámetros de las funciones o clases.

Ejemplo de uso con `help()`:

```
help(calcular_area_circulo)
```

Diferencias entre comentarios y docstrings

Característica	Comentarios	Docstrings
Sintaxis	#	""" """
Se usa en	En cualquier parte del código	En funciones, clases o módulos
Propósito	Explicar el código	Documentar cómo usar el código

Leído por help()	 No	 Sí
Visibilidad	Solo para el programador	Para usuarios y documentación automática

Conclusión

Los **comentarios** y los **docstrings** son herramientas fundamentales para escribir **código legible, mantenible y profesional**.

Un buen programa no solo debe funcionar correctamente, sino también **ser comprensible** para quien lo lea después.

Comentar adecuadamente y documentar con docstrings es una práctica que demuestra **buena calidad y responsabilidad en la programación**.