

Explicación de la Práctica 1

P1.1 – Python Basics

En este apartado realicé ejercicios introductorios con el Python. El objetivo era familiarizarse con la sintaxis básica del lenguaje y sus estructuras de control. Se practicó con variables, condicionales (if/else), bucles (for/while), listas, diccionarios y la definición de funciones simples.

Este ejercicio me dio un poco las bases necesarias para comprender la Programación Orientada a Objetos en el siguiente apartado.

P1.2 – Programación Orientada a Objetos

En este apartado trabajé con los conceptos de Programación Orientada a Objetos (POO). Cree clases con constructores (`__init__`), atributos y métodos. También practique la herencia con subclases que extienden las funcionalidades de una clase base.

Gracias a estos ejercicios se afianzaron los conceptos de encapsulación, herencia y polimorfismo, que resultaron fundamentales para el desarrollo del proyecto más completo de la P1.3.

P1.3 – Proyecto Tienda Online

En este apartado desarrollé un proyecto completo de consola que simula una tienda online. El código se organizó en carpetas siguiendo una estructura modular:

- `models/`: contiene las clases principales del dominio.
 - * `Producto` (clase base) y sus subclases `ProductoElectronico` y `ProductoRopa`.
 - * `Usuario` (clase base) y sus subclases `Cliente` y `Administrador`.
 - * `Pedido`: vinculado a un cliente y a una lista de productos, permite calcular el importe total incluyendo IVA.
- `services/`: contiene la clase `TiendaService`, encargada de gestionar usuarios, productos y pedidos de forma centralizada.
- `main.py`: archivo de prueba donde se instancia el servicio, se crean varios usuarios y productos, se listan los inventarios, se simulan pedidos y se muestra el historial de un cliente.

Mejoras añadidas respecto a la versión básica:

- Validación de que el precio y stock no sean negativos.

- Métodos renombrados para diferenciarlo de otros ejercicios (`tiene_stock`, `modificar_stock`, `crear_pedido`, `historial_pedidos_usuario`).
- Clase `Pedido` con cálculo de subtotal, IVA (21%) y total.
- Método extra `listar_usuarios()` y búsqueda de producto por nombre.
- Inventario mostrado antes y después de los pedidos.

Este apartado demuestra la aplicación completa de la POO, con herencia, polimorfismo y encapsulación, y sienta las bases para evolucionar a un proyecto con API REST en prácticas posteriores.

Conclusión

La práctica completa me permitió avanzar progresivamente desde el uso básico de Python (P1.1), a la comprensión de la POO (P1.2) y finalmente a la integración de ambos en un proyecto modular realista (P1.3). El resultado final es un sistema funcional y extensible que demuestra los conocimientos que adquirí, con un diseño cuidado y mejoras adicionales para diferenciarlo de otros trabajos.