

Programación III – Colaboración entre clases y encapsulamiento

Ejercicios obligatorios

1. Crear una clase "Empleado" que pertenecen a una institución con atributos como nombre, salario y fecha de inicio de su actividad en la institución. Implementa métodos para obtener los datos del empleado, establecer la antigüedad del empleado y calcular el aumento de salario en función de la antigüedad. La institución requiere un listado de los empleados ingresados a la institución. Definir como atributo privado el salario y definir los métodos adecuados para leerlo o bien establecer un nuevo valor.
2. Crea un sistema de tienda en línea con clases como "Producto", "Carrito" y "Cliente". Un objeto "Carrito" puede contener una lista de productos que el cliente ha agregado, y el objeto "Cliente" puede tener un objeto "Carrito" asociado. Implementar métodos para agregar clientes a la tienda, agregar productos al carrito, calcular el total de la compra y procesar el pago.
3. Crea un sistema de almacén con clases "Producto" y "Proveedor". Un objeto "Almacén" puede contener una lista de objetos "Producto", y cada "Producto" puede tener un objeto "Proveedor" asociado. Implementar métodos para: agregar productos al almacén, agregar proveedores al almacén, mostrar los productos que venden al almacén los proveedores y gestionar el inventario del almacén (altas de productos y modificaciones de precio y listado de productos sin stock).
4. Crea un sistema de correo electrónico con las clases "Usuario", "Bandeja de entrada" y "Mensaje". Un objeto "Usuario" puede tener una bandeja de entrada que contenga una lista de objetos "Mensaje". Puedes implementar métodos para enviar y recibir mensajes, así como para filtrar mensajes por categorías como leídos, no leídos, enviados, etc. Definir como atributo privado el atributo correspondiente al texto del mensaje y definir los métodos adecuados para leerlo o bien establecer un nuevo valor.
5. Crea un sistema de reserva de hotel con clases "Hotel", "Habitación" y "Reserva". Un objeto "Hotel" puede tener una lista de objetos "Habitación", y un objeto "Reserva" se relaciona con un objeto "Habitación". Implementar métodos para buscar habitaciones disponibles, realizar reservas y mostrar detalles de las reservas realizadas. Definir como atributo privado el estado de la habitación y definir los métodos adecuados para leerlo o bien establecer un nuevo valor.