

9.1. CLASES Y OBJETOS

El paradigma orientado a objetos nació en 1969 de la mano del doctor noruego Kristin Nygaard, que intentando escribir un programa de computadora que describiera el movimiento de los barcos a través de un fiordo descubrió que era muy difícil simular las mareas, los movimientos de los barcos y las formas de la línea de la costa con los métodos de programación existentes en ese momento. Descubrió que los elementos del entorno que trataba de modelar —barcos, mareas y línea de la costa de los fiordos— y las acciones que cada elemento podía ejecutar constituían unas relaciones que eran más fáciles de manejar.

Las tecnologías orientadas a objetos han evolucionado mucho, pero mantienen la razón de ser del paradigma: combinación de la descripción de los elementos en un entorno de proceso de datos con las acciones ejecutadas por esos elementos. Las clases y los objetos como instancias o ejemplares de ellas son los elementos clave sobre los que se articula la orientación a objetos.

9.1.1. ¿Qué son objetos?

En el mundo real, las personas identifican los objetos como cosas que pueden ser percibidas por los cinco sentidos. Los objetos tienen propiedades específicas, tales como posición, tamaño, color, forma, textura, etc., que definen su estado. Los objetos también tienen ciertos comportamientos que los hacen diferentes de otros objetos.

Booch¹ define un *objeto* como «algo que tiene un estado, un comportamiento y una identidad». Supongamos una máquina de una fábrica. El *estado* de la *máquina* puede ser *funcionando/parando* («on/off») su potencia, velocidad máxima, velocidad actual, temperatura, etc. Su *comportamiento* puede incluir acciones para arrancar y parar la máquina, obtener su temperatura, activar o desactivar otras máquinas, condiciones de señal de error o cambiar la velocidad. Su *identidad* se basa en el hecho de que cada instancia de una máquina es única, tal vez identificada por un número de serie. Las características que se eligen para enfatizar en el estado y el comportamiento se apoyarán en cómo un objeto máquina se utilizará en una aplicación. En un diseño de un programa orientado a objetos se crea una abstracción (un modelo simplificado) de la máquina basado en las propiedades y comportamiento que son útiles en el tiempo.

[Martin/Odell] definen un objeto como «cualquier cosa, real o abstracta, en la que se almacenan datos y aquellos métodos (operaciones) que manipulan los datos». Para realizar esa actividad se añaden a cada objeto de la clase los propios datos asociados con sus propias métodos miembro que pertenecen a la clase.

Cualquier programa orientado a objetos puede manejar muchos objetos. Por ejemplo, un programa que maneja el inventario de un almacén de ventas al por menor utiliza un objeto de cada producto manipulado en el almacén. El programa manipula los mismos datos de cada objeto, incluyendo el número de producto, descripción del producto, precio, número de artículos del *stock* y el momento de nuevos pedidos.

Cada objeto conoce también cómo ejecutar acciones con sus propios datos. El objeto producto del programa de inventario, por ejemplo, conoce cómo crearse a sí mismo y establecer los valores iniciales de todos sus datos, cómo modificar sus datos y cómo evaluar si hay artículos suficientes en el *stock* para cumplir una petición de compra. En esencia, la cosa más importante de un objeto es reconocer que consta de datos, y las acciones que pueden ejecutar.

Un objeto de un programa de computadora no es algo que se pueda tocar. Cuando un programa se ejecuta, la mayoría de los objetos existen en memoria principal. Los objetos se crean por un programa para su uso mientras el programa se está ejecutando. A menos que se guarden los datos de un objeto en

¹ Booch, Grady, *Análisis y diseño orientado a objetos con aplicaciones*. Madrid, Díaz de Santos/Addison-Wesley, 1995.