Chapter 4. Functions

# Capítulo 4. Funciones

Toda falta se condena antes de cometerse: La mía era la cifra misma de una función. . .

-WilliamShakespeare, *Measure for Measure*

Lo mejor de JavaScript es su implementación de funciones. Lo hace casi todo bien. Pero, como cabe esperar de JavaScript, no lo hizo todo bien.

Una función encierra un conjunto de sentencias. Las funciones son la unidad modular fundamental de JavaScript. Se utilizan para la reutilización del código, la ocultación de información y la composición. Las funciones se utilizan para especificar el comportamiento de los objetos. En general, el oficio de programar es la factorización de un conjunto de requisitos en un conjunto de funciones y estructuras de datos.

# Objetos de función

Las funciones en JavaScript son objetos. Los objetos son colecciones de pares nombre/valor que tienen un enlace oculto a un objeto prototipo. Los objetos producidos a partir de literales de objeto están vinculados a Object.prototype. Los objetos de función están vinculados a Function.prototype (que a su vez está vinculado a Object.prototype). Cada función también se crea con dos propiedades ocultas adicionales: el contexto de la función y el código que implementa el comportamiento de la función.

Cada objeto función también se crea con una propiedad prototype. Su valor es un objeto con una propiedad constructor cuyo valor es la función. Esto es distinto del enlace oculto a Function.prototype. El significado de esta enrevesada construcción se desvelará en el próximo capítulo.

Puesto que las funciones son objetos, pueden utilizarse como cualquier otro valor. Las funciones pueden almacenarse en variables, objetos y matrices. Las funciones pueden pasarse como argumentos a funciones, y las funciones pueden devolverse desde funciones. Además, como las funciones son objetos, las funciones pueden tener métodos.

Lo que tienen de especial las funciones es que se pueden invocar.