Chapter 3. Objects

# Capítulo 3. Objetos

Sobre un objeto casero el Amor puede guiñar el ojo.

-WilliamShakespeare, *The Two Gentlemen of Verona*

Los tipos simples de JavaScript son números, cadenas, booleanos (true y false), null, y undefined. Todos los demás valores son *objects*. Los números, las cadenas y los booleanos son similares a los objetos en el sentido de que tienen métodos, pero son inmutables. Los objetos en JavaScript son colecciones con claves mutables. En JavaScript, las matrices son objetos, las funciones son objetos, las expresiones regulares son objetos y, por supuesto, los objetos son objetos.

Un objeto es un contenedor de propiedades, donde una propiedad tiene un nombre y un valor. El nombre de una propiedad puede ser cualquier cadena, incluida la cadena vacía. El valor de una propiedad puede ser cualquier valor de JavaScript, excepto undefined.

Los objetos en JavaScript no tienen clases. No hay restricciones en los nombres de las nuevas propiedades ni en los valores de las propiedades. Los objetos son útiles para recoger y organizar datos. Los objetos pueden contener otros objetos, por lo que pueden representar fácilmente estructuras de árbol o de grafo.

JavaScript incluye una función de enlace de prototipos que permite que un objeto herede las propiedades de otro. Si se utiliza bien, puede reducir el tiempo de inicialización de los objetos y el consumo de memoria.

# Literales de objeto

Los literales de objeto proporcionan una notación muy cómoda para crear nuevos valores de objeto. Un literal de objeto es un par de llaves que rodean cero o más pares nombre/valor. Un literal de objeto puede aparecer en cualquier lugar donde pueda aparecer una expresión:

var empty\_object = {};  
  
var stooge = {  
 "first-name": "Jerome",  
 "last-name": "Howard"  
};

El nombre de una propiedad puede ser cualquier cadena, incluida la cadena vacía. Las comillas alrededor del nombre de una propiedad en un literal de objeto son opcionales si el nombre es un nombre legal de JavaScript y no una palabra reservada. Por tanto, las comillas son obligatorias alrededor de "first-name", pero son opcionales alrededor de first\_name. Se utilizan comas para separar los pares.

El valor de una propiedad puede obtenerse a partir de cualquier expresión, incluido otro literal de objeto. Los objetos pueden anidarse:

var flight = {  
 airline: "Oceanic",  
 number: 815,  
 departure: {  
 IATA: "SYD",  
 time: "2004-09-22 14:55",  
 city: "Sydney"  
 },  
 arrival: {  
 IATA: "LAX",  
 time: "2004-09-23 10:42",  
 city: "Los Angeles"  
 }  
};