Invocation

## El patrón de invocación al constructor

JavaScript es un *prototypal* lenguaje de herencia. Eso significa que los objetos pueden heredar propiedades directamente de otros objetos. El lenguaje no tiene clases.

Esto supone un cambio radical respecto a la moda actual. La mayoría de los lenguajes actuales son *classical*. La herencia prototípica es muy expresiva, pero no se entiende muy bien. El propio JavaScript no confía en su naturaleza prototípica, por lo que ofrece una sintaxis de creación de objetos que recuerda a la de los lenguajes clásicos. Pocos programadores clásicos consideraron aceptable la herencia prototípica, y la sintaxis de inspiración clásica oscurece la verdadera naturaleza prototípica del lenguaje. Es lo peor de ambos mundos.

Si se invoca una función con el prefijo new, se creará un nuevo objeto con un enlace oculto al valor del miembro prototype de la función, y this se vinculará a ese nuevo objeto.

El prefijo new también cambia el comportamiento de la sentencia return. Veremos más sobre esto a continuación.

// Create a constructor function called Quo.  
// It makes an object with a status property.  
  
var Quo = function (string) {  
 this.status = string;  
};  
  
// Give all instances of Quo a public method  
// called get\_status.  
  
Quo.prototype.get\_status = function ( ) {  
 return this.status;  
};  
  
// Make an instance of Quo.  
  
var myQuo = new Quo("confused");  
  
document.writeln(myQuo.get\_status( )); // confused

Las funciones destinadas a ser utilizadas con el prefijo new se denominan *constructors*. Por convención, se guardan en variables con nombre en mayúsculas. Si se llama a un constructor sin el prefijo new, pueden ocurrir cosas muy malas sin que se produzca una advertencia en tiempo de compilación o de ejecución, por lo que la convención sobre el uso de mayúsculas es realmente importante.

No se recomienda utilizar este estilo de funciones constructoras. Veremos alternativas mejores en el próximo capítulo.

[anterior](ch04s03_3.html)[Subtema 4 de 5: (Ver todo)](ch04s03.html)[siguiente](ch04s03_5.html)