The function Statement Versus the function Expression

# La declaración de función frente a la expresión de función

JavaScript tiene una declaración function y una expresión function. Esto es confuso porque pueden parecer exactamente iguales. Una expresión functiones la abreviatura de una expresión var con un valor de función.

La expresión:

function foo( ) {}

significa más o menos lo mismo que

var foo = function foo( ) {};

A lo largo de este libro, he utilizado la segunda forma porque deja claro que foo es una variable que contiene un valor de función. Para utilizar bien el lenguaje, es importante comprender que las funciones son valores.

function Las declaraciones están sujetas a *hoisting*. Esto significa que, independientemente de dónde se coloque un function, se desplaza a la parte superior del ámbito en el que se define. Esto relaja el requisito de que las funciones deben declararse antes de utilizarse, lo que en mi opinión conduce a la dejadez. También prohíbe el uso de declaraciones function en declaraciones if. Resulta que la mayoría de los navegadores permiten declaraciones function en declaraciones if, pero varían en cómo debe interpretarse. Esto crea problemas de portabilidad.

Lo primero que aparece en una sentencia no puede ser una expresión function porque la gramática oficial asume que una sentencia que comienza con la palabra function es una sentencia function. La solución consiste en encerrar toda la invocación entre paréntesis:

(function ( ) {  
 var hidden\_variable;  
  
 // This function can have some impact on  
 // the environment, but introduces no new  
 // global variables.  
}());