Function Literal

# Literal de función

Los objetos función se crean con literales de función:

// Create a variable called add and store a function  
// in it that adds two numbers.  
  
var add = function (a, b) {  
 return a + b;  
};

Un literal de función tiene cuatro partes. La primera parte es la palabra reservada function.

La segunda parte opcional es el nombre de la función. La función puede utilizar su nombre para llamarse a sí misma recursivamente. El nombre también puede ser utilizado por depuradores y herramientas de desarrollo para identificar la función. Si a una función no se le da un nombre, como en el ejemplo anterior, se dice que es *anonymous*.

La tercera parte es el conjunto de parámetros de la función, envueltos entre paréntesis. Dentro de los paréntesis hay un conjunto de cero o más nombres de parámetros, separados por comas. Estos nombres se definirán como variables en la función. A diferencia de las variables normales, en lugar de inicializarse a undefined, se inicializarán a los argumentos suministrados al invocar la función.

La cuarta parte es un conjunto de sentencias entre llaves. Estas sentencias son el cuerpo de la función. Se ejecutan cuando se invoca la función.

Un literal de función puede aparecer en cualquier lugar donde pueda aparecer una expresión. Las funciones pueden definirse dentro de otras funciones. Por supuesto, una función interna tiene acceso a sus parámetros y variables. Una función interna también tiene acceso a los parámetros y variables de las funciones dentro de las que está anidada. El objeto función creado por un literal de función contiene un enlace a ese contexto externo. Esto se denomina *closure*. Esta es la fuente de un enorme poder expresivo.